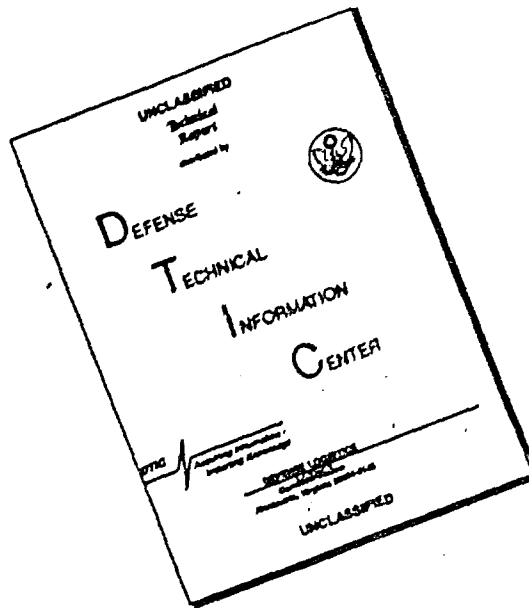


AD-A213 007

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

WHOI-89-21

**WHOI Acoustic Telemetry Project
Interim Report 12/1/88 - 6/1/89**

by

Josko Catipovic, Lee Freitag

Woods Hole Oceanographic Institution
Woods Hole, Massachusetts 02543

July 1989

Technical Report



Funding was provided by the Office of Naval Research
under contract Number N00014-86-K-0751, and
by the Charles Stark Draper Laboratory Inc.

Reproduction in whole or in part is permitted for any purpose of the
United States Government. This report should be cited as:
Woods Hole Oceanog. Inst. Tech. Rept., WHOI-89-21.

Approved for publication; distribution unlimited.

Approved for Distribution:

A handwritten signature in cursive script that reads "Albert J. Williams 3rd". The signature is written in dark ink and is positioned above a horizontal line.

Albert J. Williams 3rd, Chairman
Department of Applied Ocean Physics and Engineering

WHOI Acoustic Telemetry Project

Interim report 12/1/88 - 6/1/89

Josko Catipovic

Lee Freitag

June 13, 1989

1 Introduction and Progress Overview

This interim report covers the progress of the acoustic telemetry project during the period 12/1/88 to 5/15/89. In general, the work followed the format specified in WHOI proposal No. 5674.1. The major exception was the deletion of the transmitter array development task and a corresponding funding decrease from \$242,242 to \$170,000. In addition, the period for the funding was extended to June 30, partly due to a two month delay in project startup.

The telemetry project was centered around the construction, programming and testing of a digital receiver prototype capable of supporting future signal processing algorithms in real-time over ocean acoustic channels. The baseline receiver consists of a two-channel analog quadrature demodulator, and interface to a multiprocessor receiver for digital signal processing.

The software developed includes routines for command and control of the analog demodulator, data handling and formatting, and minimal software to digitally implement an incoherent MFSK demodulator, synchronizer and data decoder. Data storage and display programs were also completed to facilitate the performance analysis of the unit during testing.

The system was tested in Woods Hole harbor at data rates up to 4800 bits/sec. The acoustic channel was time-dispersive Rayleigh fading, and performance close to theoretical expectations was achieved. We are confident that the system error behavior is arising from channel-caused effects and known deficiencies in system performance, such as excessive synchronizer steady-state jitter.

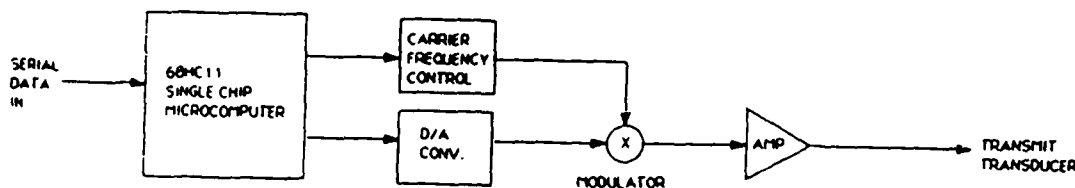


Figure 1: Acoustic telemetry transmitter block diagram

1.1 System overview

The acoustic telemetry test bed consists of a digital MFSK transmitter, developed with NSF funding, an analog demodulator, developed with ONR funding, and the signal processing receiver, developed on this project.

The transmitter is diagrammed in Figure 1. It is a small device consisting of a Motorola 68HC11 processor capable of digitally generating analog waveforms for data transmission. It is capable of transmitting 1.5 kbyte prestored buffers for experiments requiring arbitrary waveforms. It can also input a continuous data stream and modulate it in real-time at rates up to 2500 bits/sec. The system is capable of encoding the data stream with convolutional error correction codes, but this capability has not yet been used for this project. A set of software routines for programming the 68HC11 transmitter and downloading arbitrary waveforms was developed at no cost to this project and is available for use.

The demodulator is diagrammed in Figure 2. It was developed with ONR funds for other telemetry projects, and is available at no cost to this project. A printed circuit version is being constructed as a part of this project. The unit is also suitable for multichannel coherent data acquisition and processing. It consists of a piezoelectric crystal and 35 dB signal con-



tion For	
GRA&I	<input checked="" type="checkbox"/>
TAP	<input type="checkbox"/>
ounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

ditioning preamplifier housed inside a hydrophone mold. The conditioned signal is input to a two-channel surface-housed receiver. Each receiver channel consists of a digitally programmable bandpass filter with 24 dB/octave rolloff on high and low-pass ends, followed by 0 - 90 dB programmable gain amplifier, a quadrature demodulator, sin and cosine anti-alias filters with 48 dB/octave rolloff, 12-bit digitizers and an interface to the digital processor network.

The digital receiver is implemented on an INMOS T800 transputer network and represents the major development effort undertaken on this project. At present 5 processors are used. Any processor can be interfaced to the demodulator unit. In practice, one processor is interfaced to each of the two demodulator channels, and can adaptively reconfigure the demodulator as well as extract and format the input data. The other three processors perform signal processing and decoding tasks and output the data.

The transputer array is housed in a portable IBM-compatible personal computer (PC). The PC supplies the code development environment and a disk and monitor for data storage and display. The transputer compiler, linker and loader all reside on the PC disk and run within the MS-DOS operating system. This packaging yielded an easily portable system.

1.2 Tests done

The telemetry project has a permanent mini-laboratory placed at the southwest corner of the WHOI dock, and we have the use of a large harbor barge approximately 700 m away from the WHOI dock. These facilities have been used for performance testing of the acoustic telemetry system over actual marine channels. The test geometry yields a fully saturated, fading, dispersive channel, very characteristic of many shallow water and confined geometry acoustic propagation channels, and generally accepted as the worst-case ocean acoustic channel.

Acoustic data was transmitted over this test range at data rates up to 4800 bits/sec. Measurements of the channel scintillation index, impulse response, and channel transfer function have also been made. We are currently attempting to integrate those measurements into a channel model which will be used to improve the accuracy of performance prediction.

1.3 Algorithm work done

Substantial time and effort was spent writing signal processing and system software for the transputer array. At present, we have a software shell

written in the occam programming language which constitutes the receiver framework. The individual signal processing tasks are integrated into the shell, which provides inter-process data buffering and system control. A rudimentary receiver was programmed and executes in real-time at data rates up to 4800 bits/sec. The software is explained in greater detail in the transputer receiver section.

2 Description of acoustic communication test bed hardware

This section contains a detailed description of the acoustic telemetry system hardware, in particular, the digital receiver and interface to the demodulator, both of which were developed for this project. A description of the transmitter and demodulator is also included for completeness, and relevant schematics are included in an appendix.

2.1 Transmitter

The transmitter was developed for acoustic telemetry from pelagic fish, and was designed with severe space and power constraints. It employs a Motorola 68HC11 for digital waveform generation. Two on-board interrupt timers are used to control the data buffer sample rate and the duration of the data buffer transmit time. The data buffer is either constructed from sine look-up tables in real-time (at data rates less than 2.5 kbits/sec), or precomputed buffers can be read out from the data memory and transmitted. Since the 68HC11 has 2k bytes of program EEPROM, and only 500 bytes are required for the program, 1.5kbytes are available for arbitrary waveform buffers. This mode has been useful for system testing at data rates above 2.5 kbits/sec and for generating arbitrary waveform channel probe signals. A block diagram of the transmitter is in figure 1.

The transmitter generates a baseband version of the transmitted signal, consisting of up to 64 tones distributed in 128 tone bins. The baseband signal is up-shifted using a double sideband modulator. The DSB modulator replicates the baseband signal onto two adjacent bands. This provides a frequency diversity of 2 for each transmitted tone. The carrier frequency is software programmable within the range of DC - 100 kHz.

In addition to the DSB modulated data buffer, the transmitter can synthesize tone bursts useful for frame synchronization purposes. The duration

of the tone can be varied in 8 μ sec increments, and the frequency is programmable from DC - 100 kHz.

There are several operational versions of the transmitter, each suited to a different propagation geometry:

- A transmitter may be used with a 30 - 60 kHz 32 element array with a source level approx 180 dB re μ Pa. The array beamwidth is 15 degrees but can be narrowed to as little as 7 degrees by rewiring the elements. The transmitter is packaged in two 6" diameter, 20" long cylinders. The package contains a 48-hour battery pack for stand-alone remote use.
- A Tonpilz transducer with a ± 60 degree beamwidth, 10 - 40 kHz bandwidth, and source level up to 180 dB re μ Pa is also available with a 68HC11 transmitter. It is packaged in a 6" by 20" cylinder with a 150-hour power supply. This unit was used for most of system testing to date.
- Two 15-20 kHz transmitters with 180 dB power outputs are available. These units have been manufactured under license by Datasonics Inc. to our specifications. The units are standard 68HC11 transmitters with a command and control link for feedback.
- An omnidirectional 15-20 kHz version with a power output of 190 dB (peak) will be available shortly for long-range testing. This unit, also manufactured by Datasonics, was purchased specifically for use on this project.

2.2 Receiver - demodulator

The receiver processes the signals from piezoelectric hydrophones. The hydrophones currently in use consist of Benthos AQ-2 and Benthos AQ-4 piezoelectric elements coupled to a 35 dB voltage gain preamplifier. The unit sensitivity is -158 dB re μ Pa per volt at 1 kHz and 10 kHz, -159 at 30 kHz, and -161 at 100 kHz. The preamplifier drives a 50 Ohm load and is capable of driving 1 km of coax cable at 10 kHz. The hydrophone crystals and preamplifiers are electrically shielded with a coat of conducting paint, largely to reduce 60 Hz interference. The beam pattern is omnidirectional in X-Y plane up to 100 kHz and omnidirectional in the X-Z plane up to 10 kHz.

The receiver was developed as a high-capability acoustic front-end, and is specifically targeted at acoustic data telemetry and high resolution sonar. Each of the channels has the following subsystems:

- A bandpass filter is provided at the front end to avoid saturating the unit with out-of-band signals and noise. The low- and high-pass rolloff is 24 dB per octave. The center frequency and filter bandwidth are digitally programmable. Both can be set to 256 programmed positions. The selectable range can be modified by component substitution. Currently the center frequency can be set from 1 kHz to 100 kHz and the bandwidth from 1 kHz to 20 kHz.
- The bandpass filter is followed by a programmable gain stage. The gain can be adjusted over a 90 dB range in 1.5 dB steps.
- The signal is quadrature demodulated by Analog Devices AD630 demodulators. These units provide approximately 90 dB of dynamic range. The demodulating frequency is digitally selectable over a 1 - 100 kHz range, and the range can be extended by component selection.
- The quadrature demodulated signal is low-pass filtered by 48 dB per octave analog filters. Analog filters were chosen over switched capacitor units because of greater inherent dynamic range. The filters are programmable from DC to 51.2 kHz in 200 Hz steps, and the phase response is matched to better than 1 degree over the range of interest.
- The anti-alias filters are followed by simultaneous sample and holds and digitizers. Currently 12-bit 100 kHz A/D converters are used, but the components are pin compatible with 14-bit 100 kHz and 12 bit 200 kHz A/Ds.
- The A/D converters are interfaced to INMOS T800 transputers via 20 Mbit/sec serial links. A single processor controls a demodulator channel and formats the output data. The units are designed for use with a T800, although an interface to the PC AT bus is also available.

Four analog output taps are available from each channel. They are: the buffered hydrophone output, the band-pass filter output, and the two low-pass filtered quadrature channels.

The present demodulator is a wire-wrapped prototype. It exhibits approximately 1.5 bits of noise output at maximum gain. The PCB version currently in layout is expected to reduce the system noise to less than 1 bit.

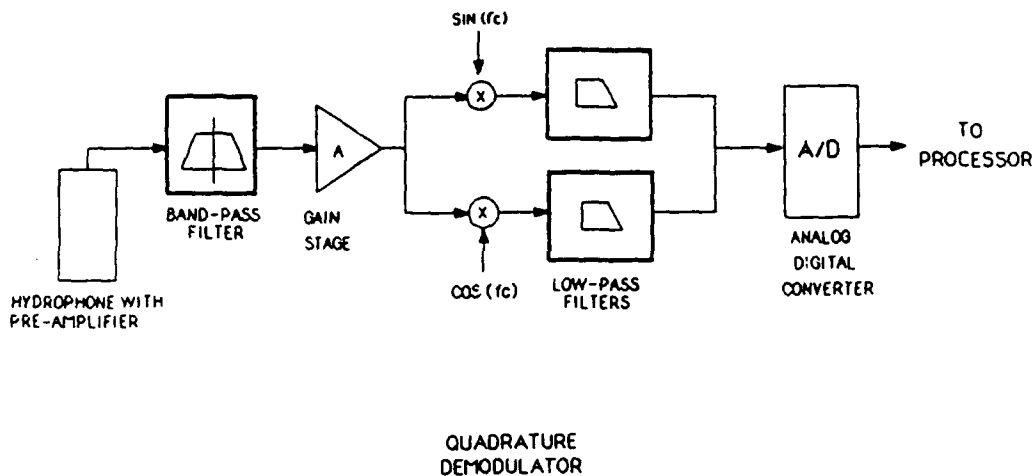


Figure 2: Demodulator block diagram

The demodulator is packaged in a 13"x12"x9" splashproof aluminum box for field use, and draws approx 30W of power. It has been used extensively on the WHOI dock and at sea.

2.3 Transputer interface hardware

A transputer serial link is a high-speed, special purpose, communication channel made for inter-processor communication. In order to talk to other types of devices, adaptor integrated circuits made by INMOS are used. These link adaptors provide eight-bit parallel I/O while maintaining proper serial link protocols at either 10 or 20 Mbits per second. Thus very high throughput may be attained with the outside world.

The programmable devices within the analog front-end are identified by eight-bit addresses, and accept eight-bit parameters. Devices take from one to three address locations depending on how many bytes are required to completely specify the program parameter. The digitized data from the A/D converters is 12-bits, and an additional bit indicates whether it is in-phase or quadrature.

The serial down-link interface puts the device address and a data value onto the digital bus of the front-end and then provides a strobe. The serial

up-link uses a data-ready signal to begin a transaction, then folds the converter output into two bytes which are passed up the link. A programmable logic device was used for a portion of the circuitry to provide a high degree of integration. The interface is housed in the same portable box as the analog circuitry. Complete schematics are included with this report.

3 Transputer-based receiver

The transputer is a powerful microprocessor developed specifically for multiprocessing and it is ideally suited for applications where the algorithms to be implemented can be mapped to a coarse-grained parallel processing architecture. The real-time acoustic telemetry receiver is one such application, and the signal processing requirements for that system map very neatly onto this architecture.

The transputer system has been programmed to work as a basic acoustic receiver processing system, and extending the system to include additional, more sophisticated, algorithms will be straightforward. The array has also been interfaced to the analog front-end described earlier in this report. The programmable devices in each channel of the front-end are accessible via transputer software, and the system is very extendable since only one link is required for each input channel and there is no common bus to saturate.

The system has been implemented on an array of INMOS T800 Transputers which are installed on an INMOS motherboard in a personal computer. The motherboard has 10 sites for transputer modules (TRAMS) and currently 5 are installed. Four of them have 32Kbytes of static RAM, and one has 128K of SRAM and 2Mbytes of dynamic RAM. This large memory TRAM is called the root transputer and it is used to run the compiler, linker and configurer. While all of the files used by the transputer software development system reside on the PC, they actually execute on the root transputer.

Transputers, their programming language and their capabilities are described in references [4], [5] and [6]. The actual hardware and software products (transputer modules, module motherboard and the development system) are detailed in manuals and INMOS technical reports. In what follows, the software development environment is described and the software written for the project to date is presented.

3.1 Developing transputer software

Developing software for the transputer network using the INMOS D705 Toolset involves several steps.

1. An occam program for each transputer must be written. This program may actually be (and usually is) a combination of several occam processes which run in parallel, each allocated processor time by a microcoded scheduler built-in to the transputer. Any one of these processes may be an occam shell which manages a program written in Fortran, Pascal or C. Thus it is possible to use conventional languages and existing code with the transputer array.
2. The program is compiled to a binary object file and linked with the appropriate libraries. For a mixed-language program, the non-occam program must be compiled and linked before the occam harness is compiled and linked.
3. After all of the programs for the array have been run through the previous step, they are ready to be configured for the network. To do this, a file which describes all of the interconnections between processors is written, and the configurer is run. The configurer puts together a file which contains all of the code for the processor array, and adds additional instructions which allow the code for each processor to be passed through the network to the correct destination.
4. To run the configured executable image on the processors a program on the PC called the alien file server is used. It loads the configured code onto the root transputer, and also takes care of requests for host resources. Thus it acts as a server and runs simultaneously on the host to provide access to files, video display, keyboard and PC memory if required.

One of the major advantages of the transputer occam system is that its parallelism is inherent, not added on. Constructs in the language allow for parallel processes to be spawned very simply, and those processes may be moved from processor to processor as required to share the load across the network. If an algorithm running with two others on a single processor overloads that processor as the algorithm is refined, it may be easily moved to its own processor with little software effort. If a particular routine exceeds the capability of a single processor, then a way of breaking it up must be

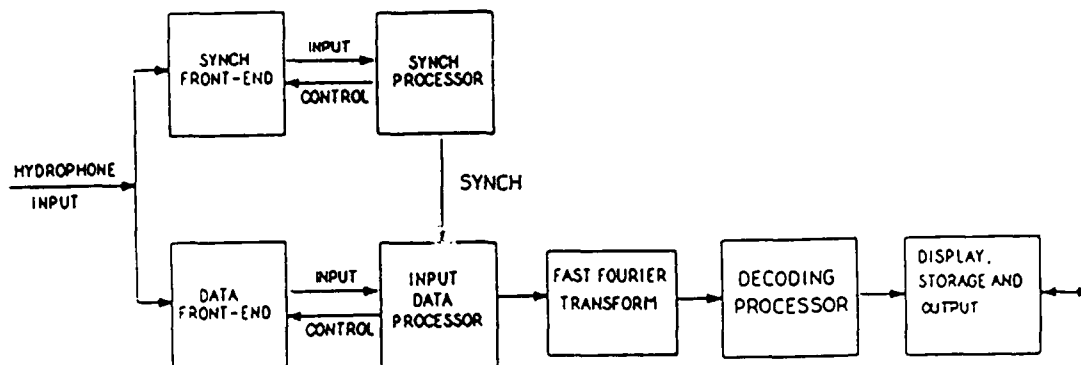


Figure 3: Transputer system block diagram

found so that it can be placed on multiple processors. This could take the form of two sequential algorithm halves, one performed after the other, or two processors working in tandem on different portions of the input data.

3.2 Receiver system

The transputer system as it is configured now includes five transputers performing the tasks as shown in Figure 3. With the exception of the transputer performing the FFT, none of the processors are heavily loaded. Since most algorithms get more complex (and thus CPU intensive) as they are developed, a conservative system configuration policy is being used to place programs within the array. This results in initial under-utilization, but will save time as program code is incrementally expanded. A detailed description of the software processes running on each transputer follows.

3.2.1 Synchronizer transputer

The synchronizer processor controls one analog front-end channel, runs an extended delay locked loop (DLL), and provides the data collection processor with an estimate of the data packet arrival time. The analog front-end is setup to band-pass filter and demodulate the received synch pulse. The filter

is programmed to pass the pulse and remove most of the data spectrum, and the demodulator brings the pulse down to baseband, in essence performing a hardware correlation. The 12-bit digitized data is sent to the DLL processor over a transputer serial link.

The synchronizer samples on either side of the synch pulse and attempts to keep the pulse in the center of the sample window. However, the data packet acquisition program must be handling data during the time just following the pulse, and is idle only after the previous packet has been received. Thus the two processes are never idle at the same time and therefore must communicate via a buffer process running in the background on the synch processor. This process simply calculates the correct start time for the packet acquisition process, and at that time sends out a "go" signal to start it. This is one of the advantages of the occam language, an intermediate process benchtop for decoupling two "orthogonal" processes is very simple and allows the connected algorithms to be as standard as possible, i.e. any idiosyncrasies in timing or format are accounted for by the buffer, and algorithms require few implementation-specific modifications.

3.2.2 Data acquisition transputer

The data acquisition transputer controls the second analog front-end channel, collects sampled data, converts it into the appropriate format and then sends it to the FFT input buffer. The control performed on the analog front-end includes programming for the correct passband and demodulation frequency and providing adaptive system gain. This automatic gain control (AGC) is done by the processor after each frame is collected. This allows gain tracking at a rate equal to the number of frames collected, which is at present approximately 80 times per second. Along with formatting the input data the program can window each data frame for the fast Fourier transform. A Hanning window is currently available, but any window generating procedure may be inserted into the program. The load for this transputer is directly proportional to the sampling rate, and as this is a parameter which should remain flexible, the entire processor was dedicated to just this one task. In addition, it allows arbitrary pre-processing if required. The current sample rate is 20 kHz complex, so data is acquired at 40 kHz. The system can run at much higher sample rates depending upon how much processing is done on each data point.

3.2.3 Fourier transform transputer

This transputer includes an input buffer and the FFT. The transform algorithm used is mixed-radix, written in occam and optimized for speed, not code size. Despite this, it fits without difficulty onto a 32K RAM transputer module. Excluding I/O it performs a 128 point complex FFT in 4.1 ms and a 256 point complex FFT in 9.6 ms, without bit-reversal.

The input buffer is a high-priority process which interrupts the FFT algorithm each time data is sent to it from the data acquisition transputer. This context switch is entirely transparent to the programmer, although the number of context switches per second was considered when this portion of the system was designed. The amount of time required for the scheduler to change tasks is 78 cycles when the FPU is in use, or 3.9 μ s, but to avoid excess switching from task to task the input data is actually sent in small bursts, several data points per burst. This makes the interprocess relationship more efficient.

3.2.4 Decoding transputer

The data from the FFT is passed to a process which recovers the information from the magnitude-square of the FFT output. At present no error correction coding is done, so this procedure simply looks at the MFSK signal and decides, based on relative magnitudes, whether a particular transmitted bit is a zero or one. To determine the number of errors in the received sequence, the incoming data is checked against the known transmitted data and the number of errors is tabulated, along with the signal-to-noise ratio.

In the future this transputer (and perhaps several others) will be used for a soft-decision, maximum likelihood, Viterbi algorithm. A short constraint length code may be implemented on a single transputer, but as the constraint length is increased, the number of M-fold comparisons required at each step increases and will eventually swamp the processor. An area of future work is a general purpose, coarse-grained, parallel version of the Viterbi algorithm.

3.2.5 Root transputer

The root transputer maintains ties with the host machine and provides access to host resources. For most programs data display and file access is all that is required. While in earlier versions of the software the decoding from MFSK and error tracking was all done on the root transputer, the current version allows root processes extensive amounts of time to write to the

screen or to disk. This effectively decouples the real-time processing portion of the transputer system from the more mundane statistic accumulation and archiving tasks.

3.3 Future transputer system work

A new, easier to use, version of the Toolset has become available, and it provides a better interface to the host computer. In putting the system described above together, many low-level routines had to be written just to do screen and disk accesses. Work will begin shortly on upgrading the system to use the new development tools, and joint work on the system shell will be done to standardize the receiver structure. This will allow work at Draper and WHOI to proceed in parallel using the same software shell and also make integration easier.

4 Test results to date

A major thrust of the project is extensive testing in the ocean environment, and the project has acquired facilities and test equipment for fast turnaround testing in Woods Hole harbor and coastal waters. We routinely set up, complete and disassemble a telemetry test with four people in four hours. This includes setting up the receiver and test equipment in a dockside van, deploying the transmitter from a coastal craft, performing the tests, and removing the equipment to the laboratory.

The first stage of the testing procedure consisted of developing a set of performance benchmarks to assure that the equipment performs close to theoretical expectations. The goal is to ensure that project hardware performs in accordance with theoretical expectations, and we assume that approaching analytic performance expectations is a necessary and sufficient confirmation of hardware performance.

The first system test was an evaluation of error probability in additive white gaussian noise (AWGN). The error probability for incoherent demodulation in AWGN is

$$Pr_e = e^{-\frac{SNR}{2}}.$$

System performance in AWGN was simulated with a programmable arbitrary waveform generator and the receiver operated in real-time to generate the results shown in Figure 4. It is seen that the system performs according to expectations. We believe this test to be significant in resolving any

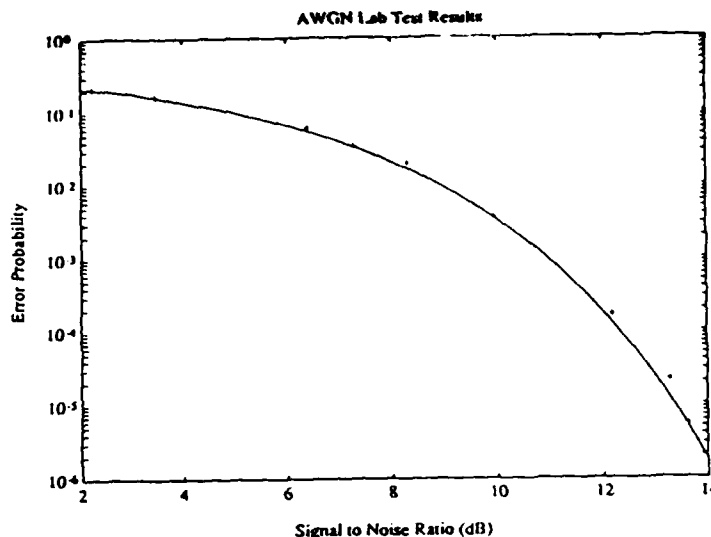


Figure 4: Telemetry system performance in benchtop AWGN environment

present and future doubts about hardware defects and breakdowns, and the test may be repeated whenever hardware malfunctions are suspected.

The second battery of tests was done at the WHOI dock. The transmitter was located near the Bigelow building and the receiver in the van at the southwest corner of the dock. Source receiver separation was approximately 30m for two tests, and 100m for the other. Three tests were run, and the results are shown in Figure 5. The scintillation index, defined by [3]

$$\sigma_I^2 \equiv \frac{E(I^2) - E(I)^2}{E(I)^2}$$

(where I is the signal intensity and $E()$ denotes the expectation operation) was measured during a short-range test and ranged between 0.3 and 0.7.

Curve 2 reflects system performance with a single diversity transmission. A single CW tone was demodulated to simulate data transmission. A CW transmission is an important performance test because it isolates the effects of signal fluctuation/fading from multipath and synchronization problems.

Curve 3 reflects error performance of a diversity 2 transmission. Performance improvement is noticeable, as expected in a fading environment.

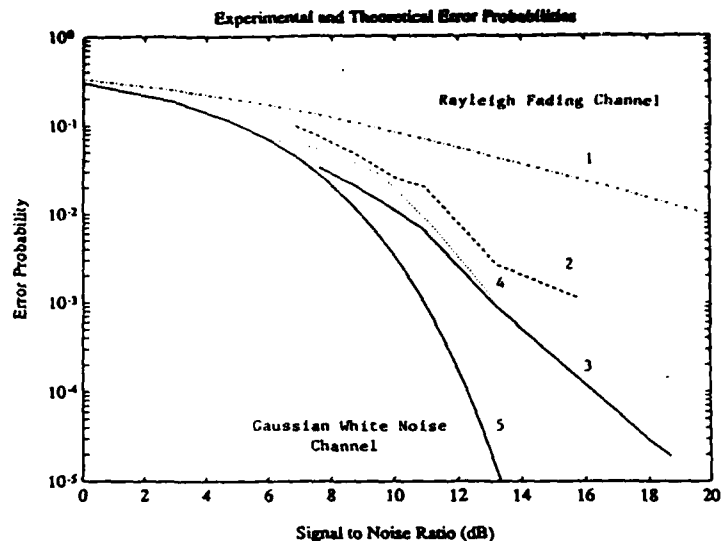


Figure 5: Telemetry system performance in a dockside experiment

Curve 4 is from an experiment involving transmission of an ASCII message string at 5 kbits/sec. The difference between 3 and 4 is due to synchronization and multipath effects. The source-receiver orientation was selected for this experiment to minimize multipath, and thus most of the discrepancy is due to errors in the synchronization process. We expect to refine this testing procedure as more work is done on the synchronization system.

Curves 1 and 5 are predicted system performance in the Rayleigh fading ($\sigma_f^2 = 1$) and the AWGN ($\sigma_f^2 = 0$) environments.

Figure 6 shows the system error performance over the 700 m path from a barge to the dock. A typical channel impulse response is shown in Figure 7. Expected system performance over the Rayleigh fading channel by a diversity 2 incoherently demodulated system is shown in the solid trace. The dotted line represents AWGN performance. It is seen that the performance is somewhat better than anticipated. We expect the discrepancies may be due to the fact that the channel may not have been fully fading, although we are still investigating possible errors arising from the SNR estimator.

We are currently developing a suite of system tests to isolate the effects of signal fading, multipath and synchronization jitter. When complete, this set of tests will allow independent optimization of synchronizer, equalizer/echo canceler and demodulator parameters, all of which are important for the

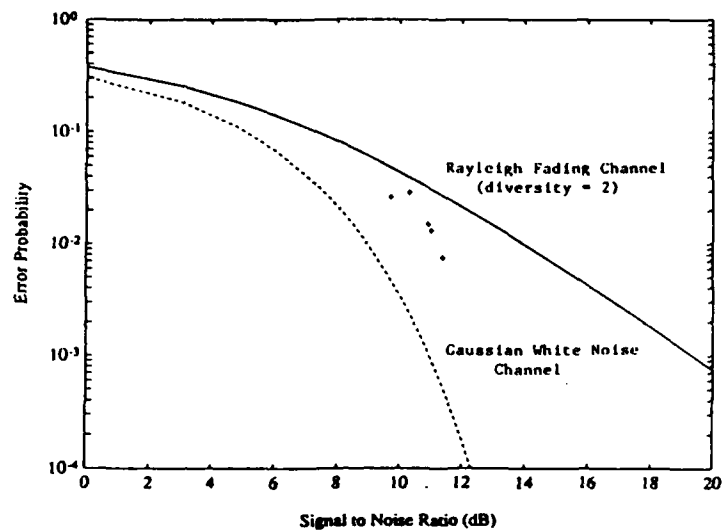


Figure 6: Telemetry system performance in a fading multipath channel

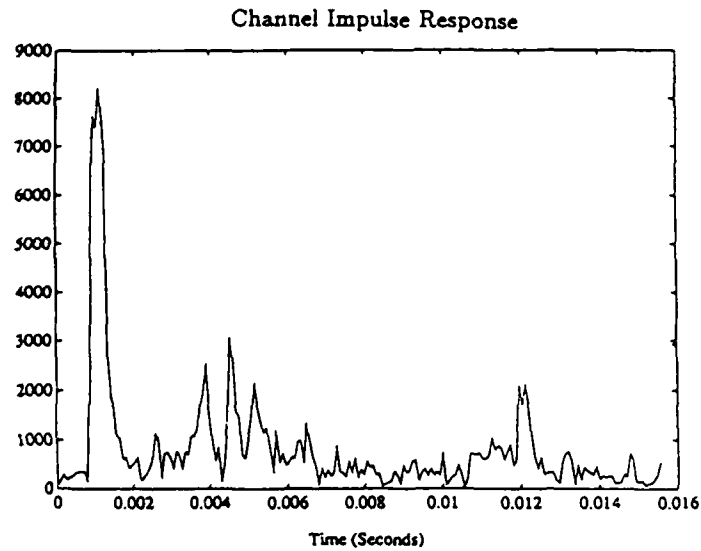


Figure 7: Typical impulse response for barge-dock transmission

next phase of development work.

4.1 Test facilities and equipment

The telemetry project has priority access to two equipment vans on the WHOI dock. One van is located adjacent to the instrument test well; this location is suitable for instrument calibrations and controlled channel experiments. We have created a controlled multipath test environment for equalization algorithm development. The multipath structure and multipath dynamics are precisely controllable, and the intent is to determine system performance in a gradually more dynamic environment.

The other van is at the southwest corner of the WHOI dock with unobstructed access to two in-harbor test ranges.

- The project has access to a harbor barge approximately 700 m away. We have access to indoor lab space on the barge. The channel between the barge and dock can be considered Rayleigh fading.
- We have established a test site at the mouth of the WHOI harbor 1.2 km away from the receiver. The site is suitable for extended transmitter deployments and was selected because of heavy current (up to 3 m/sec) and frequent boat traffic. We expect this test site to be useful for documenting system performance during extremely bad channel conditions.

The test equipment available to the project includes a signal analyzer and digital oscilloscope, both with GPIB data logging capability, and a general purpose data acquisition program written for the Transputer network. We have the capability of logging up to 400,000 complex data points from the input channel continuously, and analog recording capability using Draper Lab's VCR. Standard lab equipment is also available.

5 Future algorithm development

The acoustic telemetry system as presently configured is capable of supporting computationally intensive signal processing algorithms in real-time. This feature will be exploited during Phase II of the project with the development of error correction coding, equalization and echo-cancelling algorithms to improve the reliability of transmitted data. The development effort during Phase II will be subdivided into four tasks: error correction coding,

synchronization, least squares equalization using existing algorithms, and maximum likelihood equalization using new techniques specifically developed for the ocean acoustic channel. The four tasks will be pursued in parallel in order to make maximum use of the testing facilities and system hardware. We anticipate that it will be possible to accommodate the in-water testing requirements of all four tasks.

The following are task descriptions, personnel assignments and preliminary schedules.

5.1 Error Correction Coding

Error correction coding is required on ocean acoustic channels because of extensive signal fading. An acceptable channel model is the Rayleigh fading channel with extensive fade coherence in time, and frequency independent fading. Convolutional coding has been shown to perform well over this channel model, and we shall implement several convolutional codes to try to understand the in-water performance levels and compare them to theoretical predictions.

5.1.1 Short constraint trellis codes

Short constraint length ($K < 10$) trellis codes can exhibit significant coding gain over the Rayleigh fading channel. They are simple to implement, and the decoder complexity required for the optimal Viterbi algorithm is realizable with the Transputer network for 5 kbit/second rates. We intend to develop a simulation environment (ported over from Catipovic's thesis work) for evaluating convolutional code performance over dispersive Rayleigh fading channels. The simulation results will be compared to performance achieved over actual in-water channels.

The software development and testing effort will start in July '89, at the beginning of Phase II, and continue until December '89, the end of Phase II. We expect to implement a generalized Viterbi decoder capable of decoding any rate n/m codes. The transputer multiprocessing structure allows parallel implementations of several Viterbi decoders for real-time code comparisons over identical data sets. This task will be undertaken by Josko Catipovic and Lee Freitag at WHOI.

5.1.2 Sequential decoding of long constraint trellis codes

The error performance of short length codes degrades as fade coherence increases. While this problem is traditionally solved by signal interleaving, we believe a more robust method is sequential decoding of long constraint length codes. This method uses an extremely complex code, but the decoder complexity is reduced by taking advantage of periods of high signal quality. In a sense, the decoder "smoothes" over extended areas of degraded signal by using information from areas of high signal quality. Our theoretical analysis of algorithm performance indicates that we can expect robust performance with SNR and bit error rates equivalent to or better than any known code [1].

The software development effort will start in September or October '89 and continue at least until the end of Phase II. We expect to build on the performance results achieved with Viterbi decoding of shorter codes. This task will be undertaken by Josko Catipovic with support from Lee Freitag.

5.1.3 Binary expurgated modulation

The ocean channel is bandwidth limited by attenuation at high frequencies, and ambient noise at low frequencies. It is becoming apparent that the fundamental limits to underwater acoustic telemetry are the available bandwidth and the channel-caused signal phase fluctuation.

During Phase I, work was begun on a bandwidth-efficient integrated modulation and coding method. The method uses trellis-coded modulation of a Binary expurgated MFSK signal set. Preliminary results indicate that bandwidth expansion (BE) factors ¹ as low as 1.2 can be achieved, and 1.6 to 1.7 are practical. The code error performance is comparable to a diversity 4 uncoded system, which has a BE of 8. Uncoded MFSK has BE of 2. This technique may allow data rates of 10 kbits/sec over 10 km shallow water channels and thus extend the state of the art in acoustic telemetry.

During Phase II we intend to continue theoretical work on this technique and design and implement several test codes. We expect to have at least one code operational by the end of Phase II. This work will be undertaken by Josko Catipovic.

¹Bandwidth expansion factor is the number of Hz of bandwidth required to transmit one data bit per second.

5.2 Existing equalization algorithms

Equalization of the underwater acoustic channel is difficult because of time-variant behavior; the convergence rates of the algorithms tend to influence performance more than the steady-state behavior. Mark Fiorillo of CSDL will select and implement several promising equalization algorithms with the goal of improving overall system performance. The object is to implement reliable and known equalization methods and document their performance over actual ocean channel. This work is currently underway and we expect to have at least one operational algorithm by the end of Phase II.

5.3 EM equalization algorithms

Maximum likelihood (ML) parameter estimation, particularly estimate-maximize (EM) algorithms [7] capable of iteratively computing the maximum likelihood estimate of an extended parameter, appear very promising for equalization and channel state estimation in underwater acoustic telemetry. During Phase II, we intend to formulate the ML channel equalization problem and apply the EM algorithm.

The first task to be solved is that of optimally processing the surface echo. In many underwater telemetry scenarios, the receiver is located near the surface, and the signal echo from the surface or a nearby scatterer, such as a ship hull, can severely degrade system performance. Most past solutions consisted of physically shielding the receiver hydrophone from downward-radiating sound. This technique requires large mechanical shields and is clumsy to implement. We intend to determine the range of conditions in which optimal processing of the surface echo actually increases system performance. If the received signal is given by

$$r(t) = a_1(t)x(t) + a_2(t)x(t - T(t))e^{j\theta(t)} + n(t)$$

where $x(t)$ is the transmitted signal, $a_1(t)$ and $a_2(t)$ are time variant signal amplitudes (perhaps Rayleigh distributed), $T(t)$ is the delay difference between the two paths, and $\theta(t)$ is the phase difference between the two paths, then there exists a set of conditions where processing both arrivals improves system performance. The proposed estimator jointly estimates a_1 , a_2 , $T(t)$ and $\theta(t)$ and applies the estimates to improve the decoder. An obvious upper bound is a diversity of 2 system. Starting in July '89, we propose to quantify the performance improvement under a variety of

conditions, propose and implement an algorithm for optimal processing, and confirm our results in the water.

We will extend this work to the several-multipath case and hopefully to typical multipath structures encountered in the ocean. This work will be undertaken by Josko Catipovic of WHOI and Meir Feder of Tel-Aviv University during the summer of '89. Ehud Weinstein may also participate in this task.

5.4 Synchronization

Frame synchronization is an essential part of the telemetry system. During Phase I, an extended delay lock loop was implemented as part of the skeleton receiver. Mr. Ira Ekhous of CSDL is to undertake further synchronization system development. At the present time, frame synchronization jitter is believed to be a dominant cause of system errors, and a better understanding of the synchronization problem is urgently needed. This is a major bottleneck for system testing since robust data transmission methods cannot be readily evaluated with the present synchronizer system. We hope that a synchronization task for Phase I and Phase II will be formulated in the near future.

6 Appendix

This appendix contains a copy of the paper presented at the 6th International Symposium on Unmanned Untethered Submersible Technology by Lee Freitag. The report summarizes the work performed as part of this project, and is intended to be complementary to the rest of the interim report.

Schematics of the Datasonics-built transmitter and WHOI-built demodulator/receiver are included for completeness.

References

- [1] J. Catipovic and A. Baggeroer, "Performance of sequential decoding of convolutional codes over fully fading ocean acoustic channels," *IEEE J. of Oceanic Engineering*. Accepted for publication.
- [2] J. A. Catipovic, *Design and performance Analysis of a Digital Acoustic Telemetry System*, Sc. D. Thesis, Massachusetts Institute of Technology, 1987.

- [3] T. F. Duda, S. M. Flatte, and D. B. Creamer: *Modelling Meter-Scale Acoustic Intensity Fluctuations From Oceanic Fine Structure and Microstructure*, J.G.R. Vol. 93, No. C5, pp 5130-5142, May 1988
- [4] INMOS Limited, *Transputer Reference Manual*, Prentice Hall International, 1988.
- [5] INMOS Limited, *occam 2 Reference Manual*, Prentice Hall International, 1988.
- [6] INMOS Limited, *Communicating Process Architecture*, Prentice Hall International, 1988.
- [7] M. Segal and E. Weinstein, "Spatial and spectral parameter estimation of multiple source signals," *IEEE 1989 International Conference on Acoustics, Speech and Signal Processing*.

A Signal Processing System for Underwater Acoustic ROV Communication

Lee E. Freitag Josko A. Catipovic

Department of Applied Ocean Physics and Engineering
Woods Hole Oceanographic Institution
Woods Hole, MA 02543

ABSTRACT

A high performance system for communication with untethered underwater vehicles is presented. The system is centered around multiple digital processors which perform a variety of signal processing tasks. The processors are combined into an array using a flexible architecture designed for communication processing. A basic system has been tested at 5 kbit/sec over the Rayleigh-fading multipath channel in Woods Hole harbor.

To combat the deleterious effects of the fluctuating ocean channel, a series of algorithms designed to produce low error-rate communication is being implemented on the array. The processing elements are INMOS T800 transputers and the number of transputers may be selected at deployment time to meet the requirements of a particular task. The result is a powerful and flexible architecture for underwater acoustic communication.

INTRODUCTION

One of the many difficult tasks facing designers of unmanned untethered vehicles operating in the ocean is providing reliable, high-speed, communication with the remote vehicle. A focus of current research is acoustic telemetry in the 10 - 50 kHz band. While the range of acoustic signals at these frequencies should in theory yield communication links with rates of 5k to 20k bits per second (bps), attaining theoretical performance limits is difficult due to the complexity of the required processing system.

The environment that an underwater vehicle may be operated in may change rapidly depending on sea conditions and channel geometry. Some of the problems frequently encountered are signal fading, phase fluctuation and multipath, and all of these processes are time, frequency and space dependent [1] [2] [3]. Overcoming the effects of these phenomena requires sophisticated signal processing algorithms and a processing engine capable of implementing them in real

time. The signal processing system described here is designed to allow for the implementation of algorithms which will make possible communication under worst-case conditions.

The project described in this paper has two main goals: develop theory and techniques for high-speed digital communication in the ocean by acoustic means and develop signal processing hardware and software for real-time, in-water, testing of algorithms using a multiprocessing approach. Theoretical work is centered on counteracting multipath, fading and Doppler as well as providing synchronisation between source and receiver. The goals for the processing portion of the work include the general task of developing multiprocessor architectures for acoustic telemetry, and the specific work involved in developing a system capable of real-time signal processing.

This paper is divided into three main sections. The first describes the problems faced by an acoustic telemetry system designed to work in the ocean. The second contains background on the multiprocessor system, introduces the notion of a relocatable process and discusses methods for connecting processes. In the third section the complete system and results from one test are presented.

PROBLEM FRAMEWORK

The problems which the signal processing system is being used to overcome are described below. In addition, the communication scheme selected for implementation is presented and the type of algorithms required to effectively process the transmitted data are discussed in terms of specific acoustic channel phenomena.

Acoustic telemetry in the ocean represents a standard, though quite difficult, communication problem. In the design of any communication link the transmission medium and link requirements define the resulting system. In what follows, the ocean is discussed as a communication channel, and the dif-

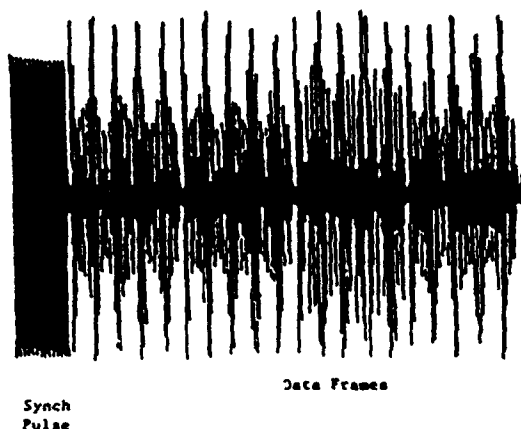


Figure 1: Time waveform with synch pulse.

difficulties associated with using that channel are described. Means of overcoming channel difficulties are suggested, and a communication system framework is presented.

Modulation and transmission

In selecting a modulation method for the worst-case ocean channel the issue of phase stability was considered carefully. While coherent communication may be possible under some conditions, an incoherent modulation technique, multiple frequency shift-keying (MFSK), was chosen for simplicity and robustness [4]. MFSK is an M-ary version of standard FSK. The number of bits which may be transmitted at any given time depends on the frequency spacing Δf and available bandwidth. If one bit is represented by a tone in one of two frequency bins, and a spacing of 78 Hz is used, then 64 bits requires 10 kHz of bandwidth. The amount of time that a set of tones must be transmitted is $1/\Delta f$ and thus for 78 Hz spacing frame duration is 12.8 msec. The corresponding data rate is 5k bps.

A frame may be generated using an inverse FFT or a sine lookup table, depending on the number of FSK tones used. An inverse FFT is efficient for a large M ($M > 16$), and a prestored table if M is small. After modulation to MFSK, this baseband signal is up-shifted to the desired carrier frequency. Single or double sideband modulation (DSB) may be used, but DSB modulation doubles the number of transmitted baseband tones, in effect implementing explicit frequency diversity for each transmission. In the Rayleigh fading environment this method is valuable, even if error correction is used [1].

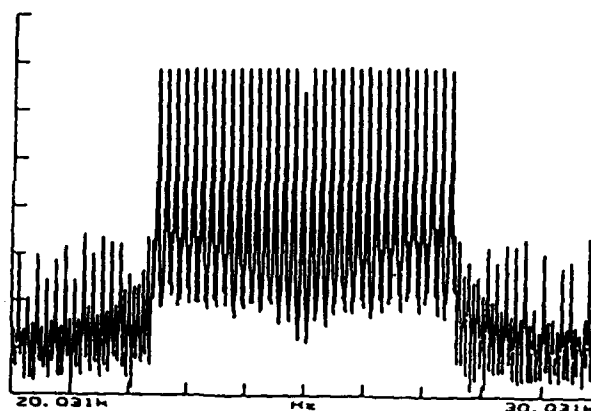


Figure 2: Spectrum of the time waveform showing both sidebands centered around the modulation frequency.

Figures 1 and 2 show examples of a signal in time and frequency. The time waveform begins with a synchronization pulse and it is followed by a data packet made up of a number of data frames. The spectral plot shows the arrangement of tones around the carrier frequency.

Synchronization

Incoherent demodulation of the MFSK sequence described above requires an external synchronization reference to indicate the start of a data frame. There will normally be movement between source and receiver so that packet arrival time varies continuously. Thus, new arrival estimates must be made frequently. One reason for selecting the incoherent MFSK system is that it eases synchronization requirements. In current work frames of 12.8 msec are used, and it has been shown that significant high frequency channel fluctuations are smoothed over by extending a tone duration past 10 msec [2].

The synchronization task for MFSK packet transmission reduces to the problem of accurately tracking the arrival of each packet. The required synch accuracy is proportional to the data frame duration, and is typically 10% of the frame length [5]. One method of synchronization is the delay-locked loop (DLL) which performs optimally in the additive white gaussian noise (AWGN) environment [6]. However, signal fading and multipath both influence the performance of the DLL synchroniser. The mean time to lose lock is particularly affected [5] [7], and multipath causes a quasistatic offset in the desired synchroniza-

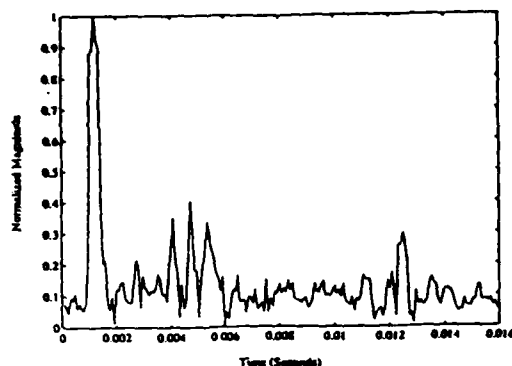


Figure 3: Initial arrival and reflections due to multipath of a 300 microsecond pulse transmitted across Woods Hole harbor.

tion estimate [6]. Reducing the steady state synchronizer jitter, minimizing the mean time to lose lock in a fading environment, and providing consistent synch estimates in the presence of time-varying multipath are currently being investigated.

Fading and error correction

The dispersive nature of the ocean medium is manifested in frequency dependent signal fading [3]. Communication over a fading channel requires a significantly larger signal-to-noise ratio (SNR) than a channel with only additive white gaussian noise. The shallow-water, long-range, horizontal channel at frequencies above 15 kHz may be modeled as Rayleigh, and the uncoded bit-error probability is given by

$$Pr_e(SNR) = \frac{1}{2 + SNR} \quad (1)$$

For $Pr_e = 10^{-6}$, 60 dB of SNR is required. Convolutional coding is one method of improving performance over correlated fading channels [8] [9]. Because convolutional codes with rates of less than 1/2 significantly reduce total link throughput in an already bandwidth limited system, work is being done on more bandwidth efficient coding methods.

Multipath and echo canceling

As an acoustic signal propagates through the water, reflections due to local geometry and temperature gradients cause it to be seen at the receiver as a distinct series of arrivals, each the result of travel

over a different path. Figure 3 shows the arrivals of a 300 microsecond pulse transmitted over 700m in Woods Hole harbor. A group of secondary signals appears about 3 msec after the initial pulse, and another approximately 12 msec after the initial arrival.

Multipath such as that shown in Figure 3 causes a data frame to be corrupted with interference from previously sent frames arriving late. Inter-symbol interference resulting from multipath must be compensated for with some knowledge of the channel impulse response, and the impulse response changes naturally over time and due to shifts in source-receiver geometry. A system for dealing with echoes must monitor changes in the impulse response and track it as it varies. Many methods exist for canceling echos in a time-varying environment [8], but the speed with which the impulse response can change makes this problem particularly challenging.

Doppler shift

The result of a non-zero relative velocity between source and receiver is a Doppler shift in the received data spectrum given by

$$f_r = f_s \frac{c - v_r}{c - v_s}$$

where c is the velocity of propagation, v_r and v_s are the receiver and source velocities, and f_s is the transmitted frequency.

MFSK modulation uses signal tones Δf Hz apart. While $|f_r - f_s| \ll \Delta f$, Doppler effects are insignificant, but as $|f_r - f_s| \sim \Delta f$, a Doppler correction is required. Doppler may be estimated from a dedicated pilot tone using a narrowband phase-locked loop. The loop bandwidth is determined by the maximum expected Doppler and the Doppler rate. An earlier version of the system employed Doppler tracking with good results [10]. Correction for Doppler may be implemented through carrier frequency and data sample rate adjustments. The first removes the shift at the demodulation frequency, and the latter the differential between frequencies across the spectrum.

A communication system

In discussing the physical realities which affect underwater data transmission, potential solutions in the form of modulation methods and specific signal processing algorithms have been identified. The system, shown in Figure 4, includes blocks which address each of the sections outlined above. Before the signal processing system that has been developed to implement Figure 4 is presented, the processor elements and real-time processing are discussed.

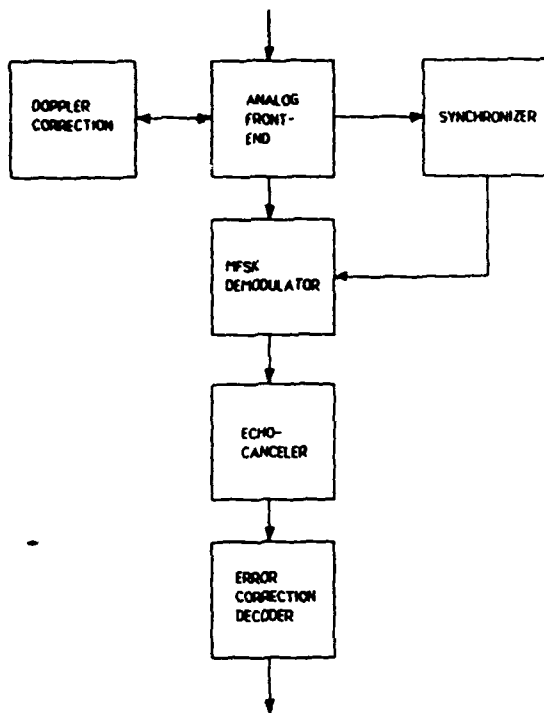


Figure 4: Diagram of the communication system.

PROCESSING FRAMEWORK

The elements chosen for the processor array are INMOS T800 transputers [11]. The T800 has a floating point unit, built-in multitasking capability and four high-speed serial links for interprocessor communication. With multitasking, one or more algorithms may be run concurrently on a single processing element.

The language developed for parallel processing using transputers is occam, but other programming languages such as Fortran, Pascal and C may be used with it. The transputer and occam form a powerful combination for multiprocessing because they allow the definition of processes and their interconnections independent of the final physical configuration [12].

Processes

A process is a computational block with defined input and output. The simplest process does nothing except I/O. Processes, which may be made up of any number of procedures and functions, communicate over predefined channels which may be inter- or intra-processor. Multiple processes can run on one element up to the computational limit of the processor and moving processes from one processor to another is not difficult.

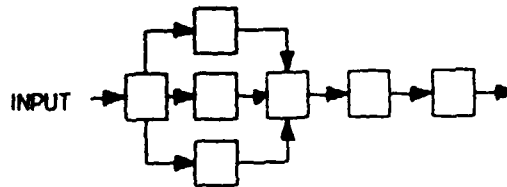


Figure 5: Array for combined parallel and pipelined processing.

The flexibility of this approach becomes apparent when the computational load of a given process changes during some stage of system development. Processes can be easily redistributed to even the load across the array when required.

Speeding things up

While in many cases algorithms will run sufficiently fast on one processor, some real-time signal processing tasks will require several processors. These algorithms may be run on a group of processors serially, using a pipeline of processors, or in parallel, where execution of independent tasks can take place simultaneously. The method used depends on precedence and whether or not the algorithms are parallelizable. An example of a combined system is shown in Figure 5. The left-hand portion of the array is used to speed up a task which can be broken up into pieces that may be executed simultaneously. The remainder of the processes require the output of the previous one before beginning.

Connecting processes

A concern in any signal processing system running on an array of processors is interconnection of algorithms with different data flows. There are two types of tasks normally encountered in signal processing, block-type tasks such as the FFT, and stream-type tasks such as filtering [13]. While a filter can perform an iteration each time a new data point arrives, a block-type procedure such as an FFT needs the entire data set before it can begin. Efficient connection of two different types of processes necessitates controlled buffering to decouple the processes. Buffer use allows removal of strict synchronization requirements: data is sent to a buffer as it is processed, and the next process gets it from the holding buffer as it needs it.

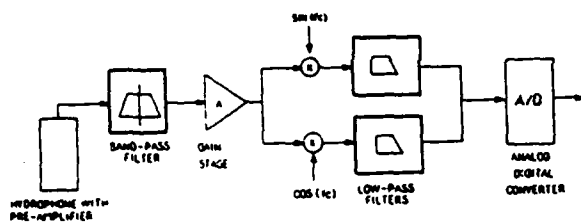


Figure 6: Standard analog front-end channel. Band-pass filter, gain, demodulation frequency, lowpass filter and sampling frequency are all programmable.

Input and output

To construct a complete system some means of getting data in and out of the array is needed. Transputer communication links are specialized, but interface logic is straightforward and a variety of products which allow transputers to be connected to standard serial or parallel devices and mass storage units are commercially available.

To get data into the processor network a standard programmable analog front-end (Figure 6) has been designed and interfaced to a transputer serial link. Bidirectional communication allows control over those devices in the front-end which are programmable, and this makes things like digitally controlled amplifier gain possible. The unit has been designed for flexible use in acoustic telemetry research, and multiple units may be used for multichannel applications such as beamforming. The decentralized, one-channel, one-processor approach is very powerful for high-speed acquisition and signal processing because additional processing elements may be added in layers as required to avoid saturating any portion of the processor network.

Power, speed and size

The transputer is available in modules with different amounts of memory. The smallest module has 32K of fast static RAM, consumes approximately 1.2 watts, and measures 3.7x1 inches. A 2-Mbyte, 5-watt module is only 3.7x2.15 inches. The T800 transputer running at 20 Mhz performs 10 million instructions per second, and 1.5 million floating point instructions per second. It does 4000k single length Whetstones per second as compared to the VAX 11/780 FPA running UNIX 4.3 BSD which attains 1083k Whetstones per second [14]. The compact size of the units, their

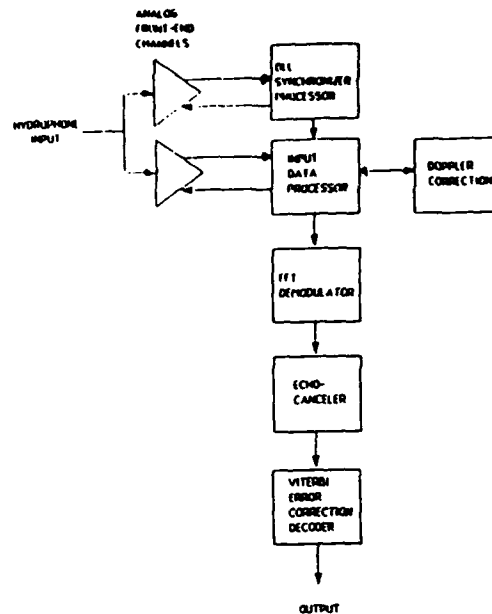


Figure 7: The transputer-based telemetry system.

low power consumption and high performance make them well suited for real-time multiprocessing.

A REAL-TIME ACOUSTIC TELEMETRY SYSTEM

An array of transputers has been used to put together the framework of a complete system capable of handling the underwater communication problem described in the first section of this paper. A block diagram of the system is in Figure 7, and it may be seen, by comparing this figure with Figure 4, that the processor solution matches very closely the communication scenario as described earlier. Each algorithm is a process which may be placed on its own processor, or with others, if it does not require the resources of an entire processor. In what follows, each of the blocks in Figure 7 is discussed.

Synch subsystem and data collection

The synch processor runs an extended delay locked loop algorithm [15] in real-time. It uses the pulse transmitted at the beginning of a data packet to provide the data collection unit with information on the exact start of the signal waveform. A programmable analog front-end is used to separate the synch pulse from the data and demodulate it to baseband. This

is then digitised and provided to the DLL algorithm. The demodulator performs a hardware correlation, but the remainder of the algorithm is done in software.

The algorithm locks and then tracks by examining the energy in a window of sampled data containing the pulse. In its simplest form this algorithm is not computationally intensive, but a complex version (capable of producing fine estimates with low jitter) requires additional processing power. For this reason the synchronizer has its own processor.

The synch processor has complete control over the analog channel it is connected to, and the output of this processor is a message to the data collection processor as to when to begin collecting the current packet. This subsystem can be considered a "black-box" by the rest of the system, and the algorithm running on it may be changed without modifying the rest of the system, as long as the form of the output, an arrival estimate, stays the same.

The analog units are separate and programmable, so both processors exercise independent control over signal conditioning, demodulation and sampling parameters. This feature will allow the incorporation of spatial diversity at the receiver by replicating the data receiver front-end and the associated local processing.

MFSK demodulation

The received data is passed to a processor which performs Fourier transforms on each frame. A mixed-radix FFT program which can do a 256-point complex FFT in less than 10 ms is used. If higher speeds are required, the mixed-radix technique allows breaking the algorithm into two pieces, each of which can be run on a different processor. As with the synchronizer subsystem above, the FFT subsystem, no matter how many processors it requires, can be considered a single unit and treated accordingly.

Doppler correction

Correction for frequency errors due to movement between source and receiver can be done by adjusting the demodulation frequency and the sampling rate. A processor digitally calculates the Doppler shift of the incoming signal, then utilises the programmability of the demodulator and the sample frequency generator to compensate. These two parameters can be set in small but discrete steps, and the object is not to track them exactly, but to keep $|f_r - f_s|$ less than some Δf which does not affect system performance. The spacing of the MFSK tones may be selected so that several knots of relative motion does not affect the error rate.

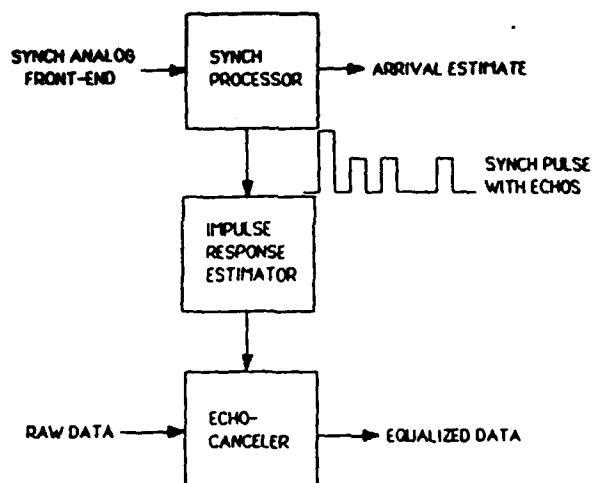


Figure 8: Echo-canceler using an impulse response estimate from the synch pulse.

Error correction decoding

To cope with fading and other error-inducing channel mechanisms convolutional coding has been selected for implementation. Convolutional coding coupled with a Viterbi decoder is a means of error control which encodes data into a semi-infinite stream then, upon reception, decodes it using a sliding window across a trellis which represents the possible states of the encoder [9]. The amount of redundancy introduced into the data stream governs the effectiveness of the code and this in turn affects the complexity of the decoder. The Viterbi algorithm uses soft-decision decoding, so the output of the FFT may be provided to it directly, without any intermediate processing.

Echo canceling

The process flow for a basic echo-canceler is shown in Figure 8. This scheme uses the synch pulse to generate an estimate of the channel impulse response. The impulse response estimator provides the relative amplitudes and time delays to the echo canceler, which then compensates for the addition of the delayed replicas. Note that the impulse response is updated after each synch pulse. We are currently investigating ML sequence estimation equalisers and an integrated equaliser-decoder.

SYSTEM PERFORMANCE

A real-time acoustic telemetry receiver using a portion of the system described above has successfully received data at 5k bps over 700 meters in Woods

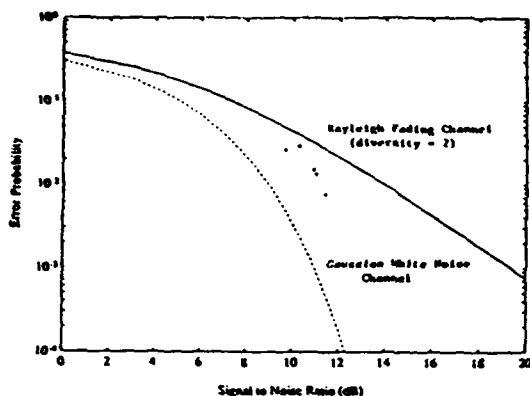


Figure 9: Error probabilities for Woods Hole harbor experiments, 700m range in water 6-18m deep.

Hole harbor. Both source and receiver were fixed, the transmissions were uncoded, and echo canceling was not implemented. The tests have used a transmitter developed at WHOI for remote ocean instrumentation. For research purposes it has been programmed to send prestored buffers. The results of one 5k bps test are shown in Figure 9. This test was conducted at 25 kHz, and used data frames 12.8 milliseconds long, each containing 64 bits of information. The upper and lower curves in Figure 9 are theoretical error probabilities for incoherent communication in the Rayleigh fading channel (with diversity two), and the additive white gaussian noise channel. Additional experiments have yielded error rates as low as 0.003 over the same path.

Error correction coding and echo-canceling algorithms are being evaluated and tested, and processor blocks in the array are reserved for those algorithms. As the algorithms are tested in the water, additional processors will be added to reach the desired performance level.

SUMMARY

This project's work in acoustic telemetry has centered on developing a flexible signal processing system and on algorithms for improving the reliability of underwater communication. Using this processing system algorithms will be tested in real-time and under realistic conditions. Theoretical work will be transferred to the system as it is developed, and iterations that require additional processing power can be easily accommodated.

ACKNOWLEDGEMENTS

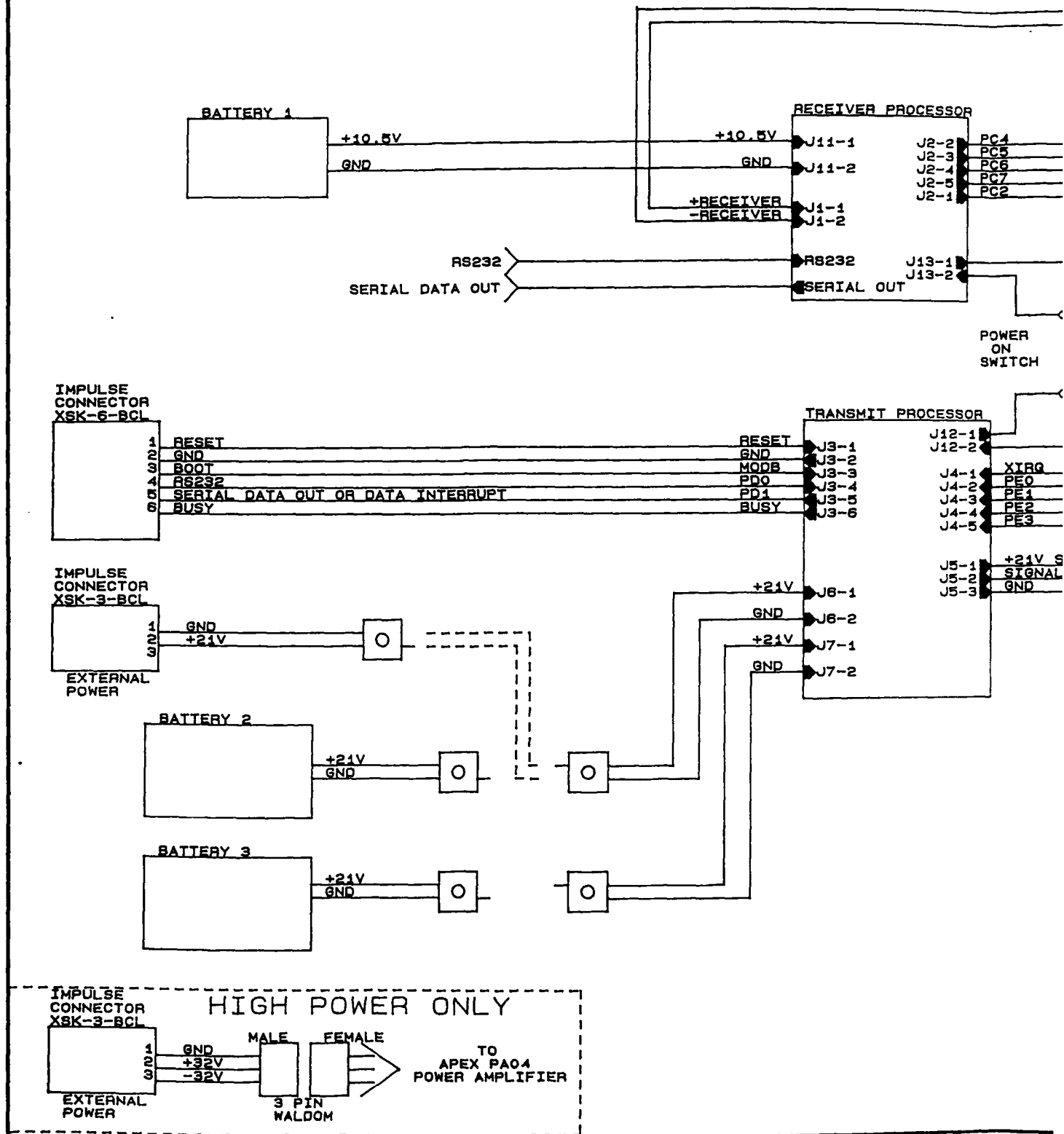
This work was supported by the Charles Stark Draper Laboratory and by the Office of Naval Research under Grant N00014-86-K-0751.

This is WHOI contribution number 7116.

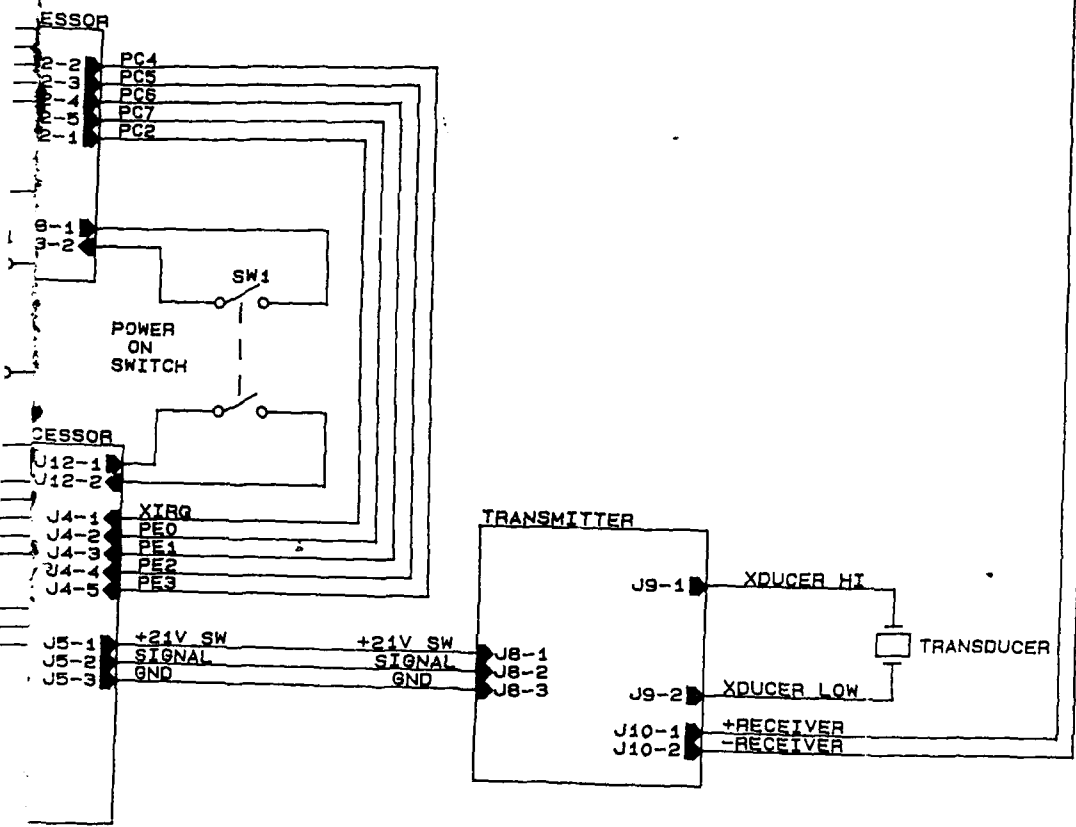
References

- [1] J. A. Catipovic, *Design and performance Analysis of a Digital Acoustic Telemetry System*, Sc. D. Thesis, Massachusetts Institute of Technology, 1987.
- [2] T. F. Duda, S. M. Flatte and D. B. Creamer, *Modelling Meter-Scale Acoustic Intensity Fluctuations From Oceanic Fine Structure and Microstructure*, J. G. R. Vol. 93, No. C5, pp 5130-5142, May 1988.
- [3] S. M. Flatte, ed., *Sound Transmission Through a Fluctuating Ocean*, Cambridge University Press, 1979.
- [4] J. A. Catipovic and A. B. Baggeroer, *Analysis of High Frequency Multitone Transmissions Propagated in the Marginal Ice Zone*, J.A.S.A., to appear.
- [5] J. A. Catipovic and L. E. Freitag, *Performance of Incoherent FFT Demodulators in the Presence of Synchronization Errors and Doppler Shifts*, in preparation.
- [6] J. K. Holmes, *Coherent Spread Spectrum Systems*, Wiley, 1982.
- [7] S. S. Soliman and R. A. Scholtz, *Synchronization Over Fading Dispersive Channels*, IEEE Trans. on Communications, Vol. 36, No. 4, April 1988, pp 499 - 505.
- [8] J. G. Proakis, *Digital Communications*, McGraw Hill, 1983.
- [9] G. C. Clark Jr. and J. B. Cain, *Error-Correction Coding for Digital Communications*, Plenum Press, 1981.
- [10] J. Catipovic, A. B. Baggeroer, K. von der Heydt and D. Koelsch, *Design and Performance Analysis of a Digital Acoustic Telemetry System for the Short Range Underwater Channel*, IEEE Journal of Oceanic Engineering, Vol OE-9, No. 4, pp. 242-252, October 1984.
- [11] INMOS Limited, *Transputer Reference Manual*, Prentice Hall International, 1988.

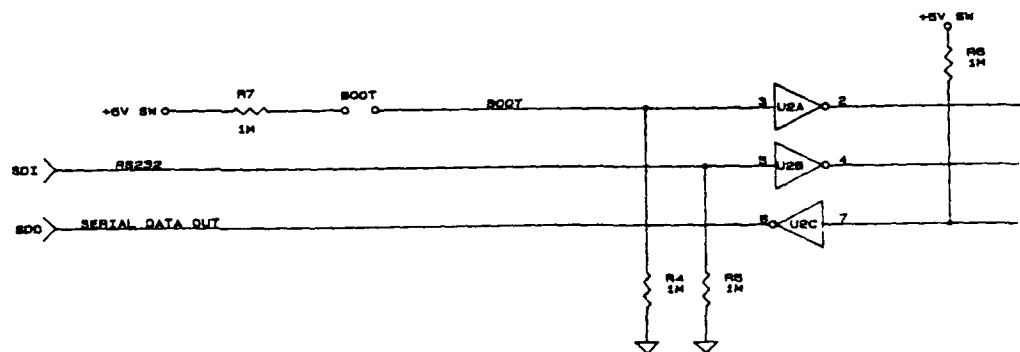
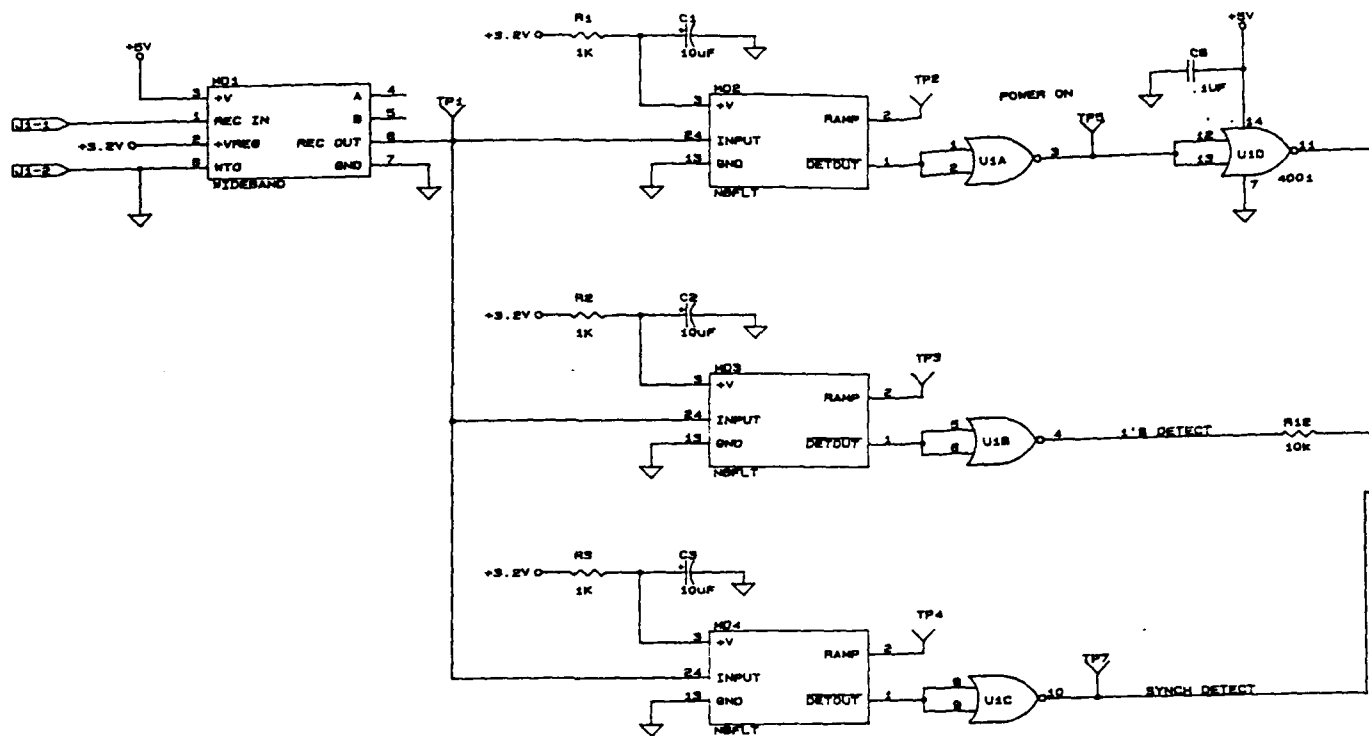
- [12] INMOS Limited, *occam 2 Reference Manual*, Prentice Hall International, 1988.
- [13] K. Konstantinides, R. T. Kaneshiro and J. F. Tani, *Scheduling and Task Allocation for Parallel Digital Signal Processing Architectures*, IEEE International Conference on Acoustics, Speech and Signal Processing, Glasgow, 1989.
- [14] INMOS Limited, *Communicating Process Architecture*, Prentice Hall International, 1988.
- [15] W. M. Bowles, *Correlation Tracking*, Sc. D. Thesis, Massachusetts Institute of Technology, 1981.



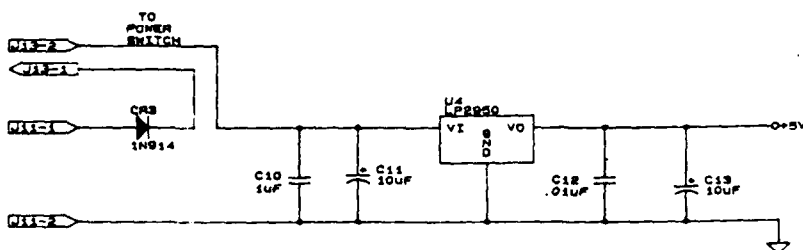
DATE	SYM	REVISION RECORD	AUTH.	DR.	CK.



TOLERANCES (EXCEPT AS NOTED)		<i>datasonics</i>	
DECIMAL ± ~	SCALE ~		
FRACTIONAL ± ~	DRAWN BY JBD		
ANGULAR ± ~	APPROVED BY		
TITLE CAREY MODEM WIRING DIAGRAM		DATE 6-14-90	
DRAWING NUMBER BS30-03432		31	



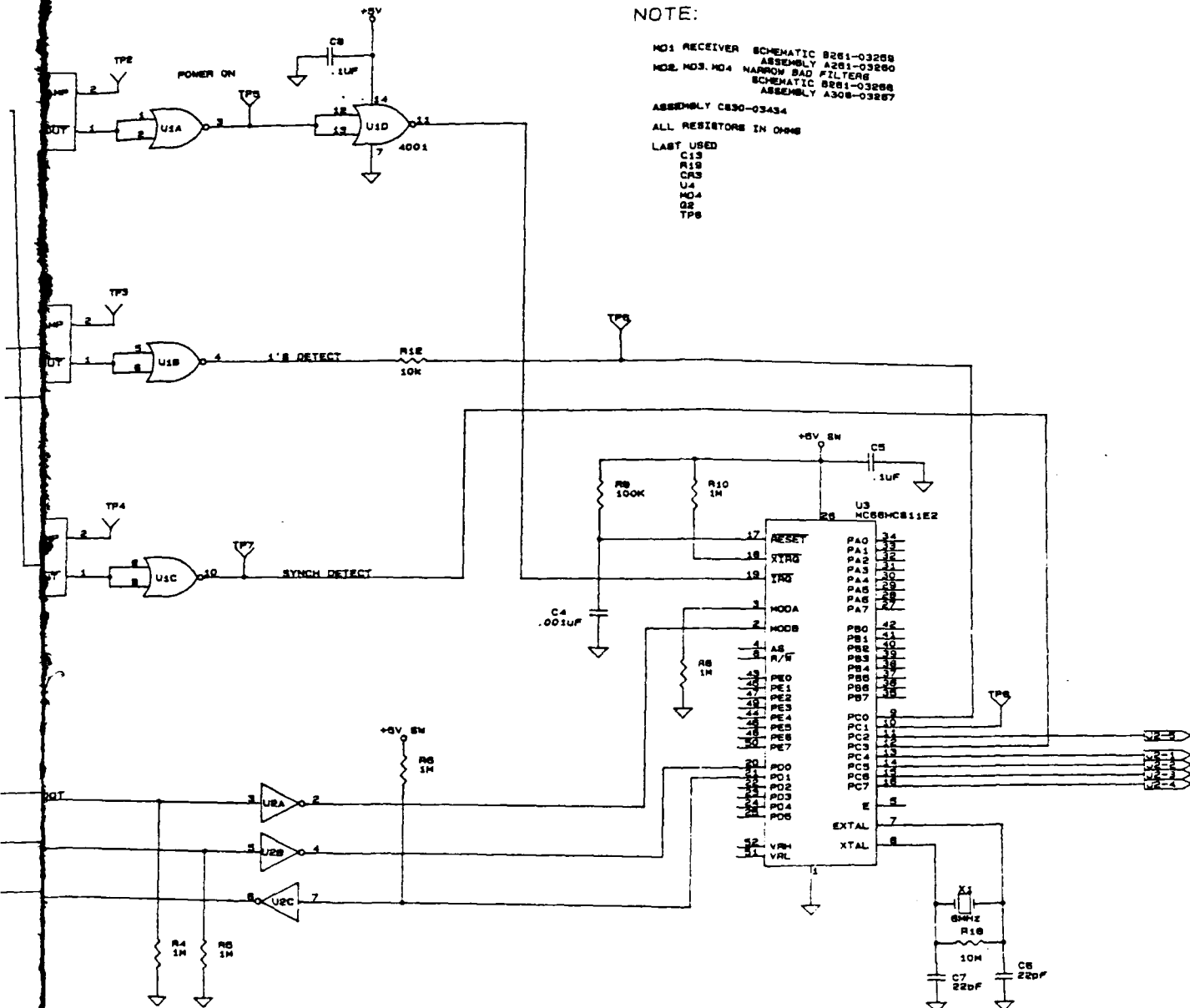
UNUSED GATES



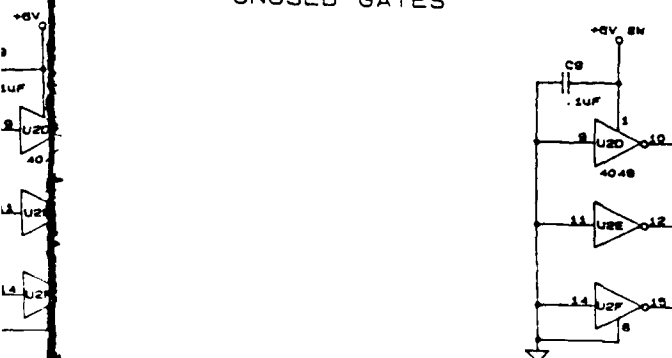
DATE	BY	REVISION RECORD	AUTH	OR	CR

NOTE:

NO1 RECEIVER SCHEMATIC 8281-03258
 ASSEMBLY A281-03259
 NO2, NO3, NO4 NARROW BAND FILTERS
 SCHEMATIC 8281-03266
 ASSEMBLY A308-03267
 ASSEMBLY C830-03434
 ALL RESISTORS IN OHMS
 LAST USED
 C13
 R19
 C23
 U4
 MO4
 Q2
 TP8



UNUSED GATES



TOLERANCES EXCEPT AS NOTED		datasonics	
DECIMAL		SCALE	DRAWN BY
FRACTIONAL			APPROVED BY
ANGULAR	DATE	DRAWING NUMBER	
		32	

NOTE:

1. MULTI-RECEIVER MODULE

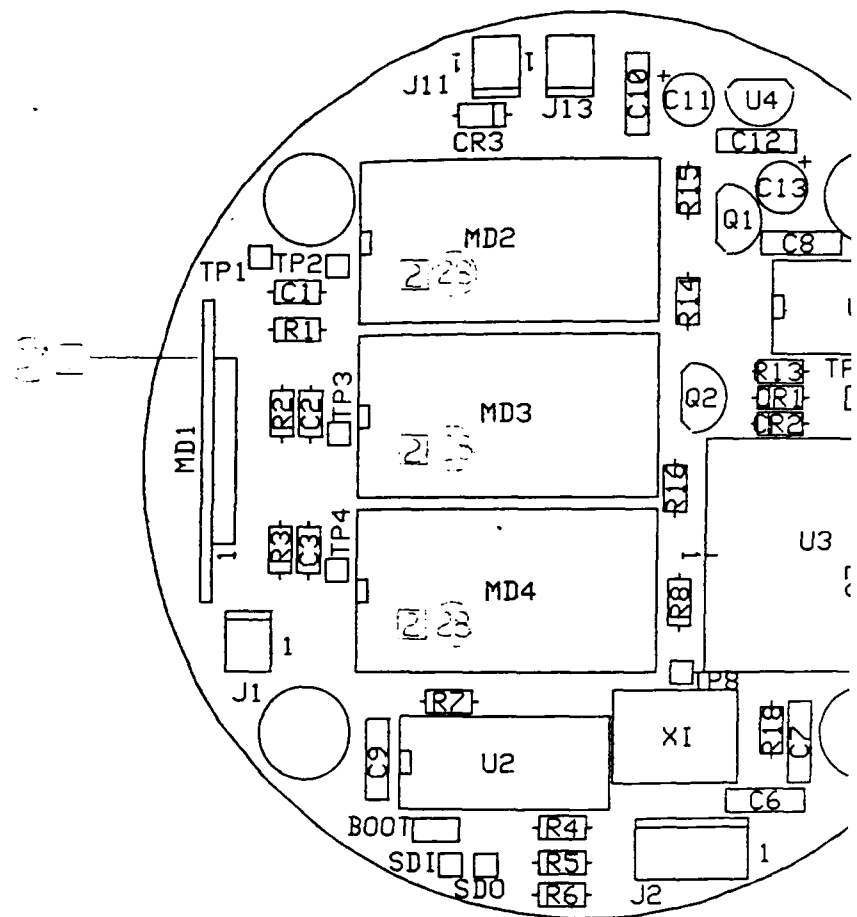
SCHEMATIC ID: A200

ASSEMBLY: A200-C332-

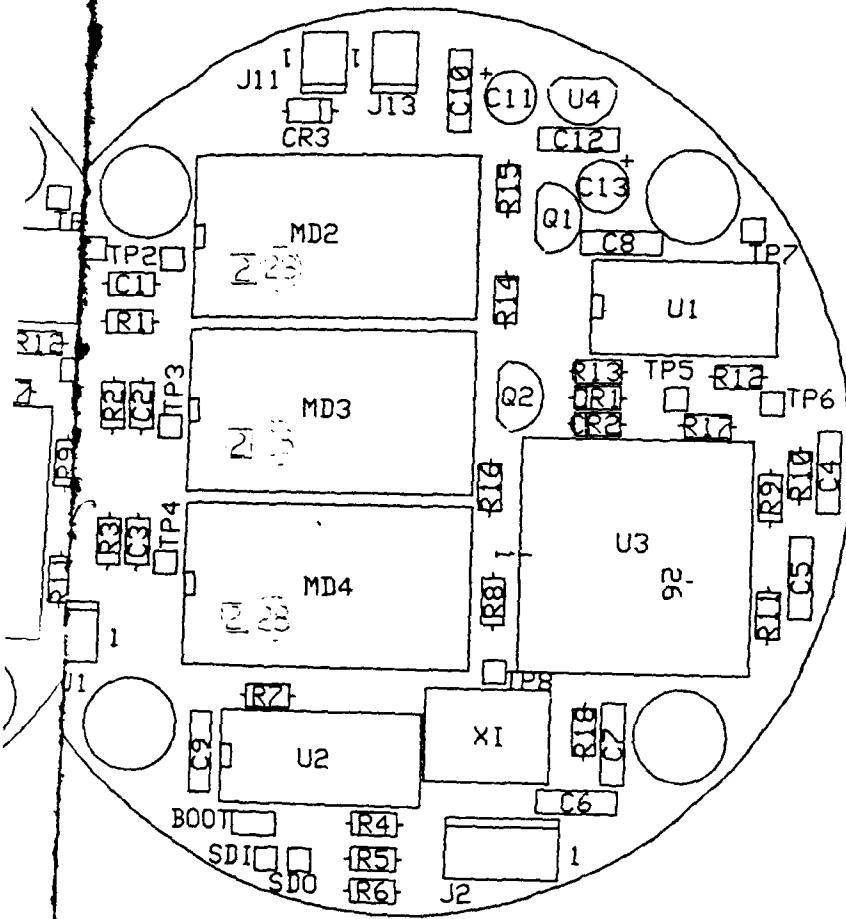
2. 64-KHZ BAND FILTER

SCHEMATIC ID: A200-C326-

ASSEMBLY: A200-C326-

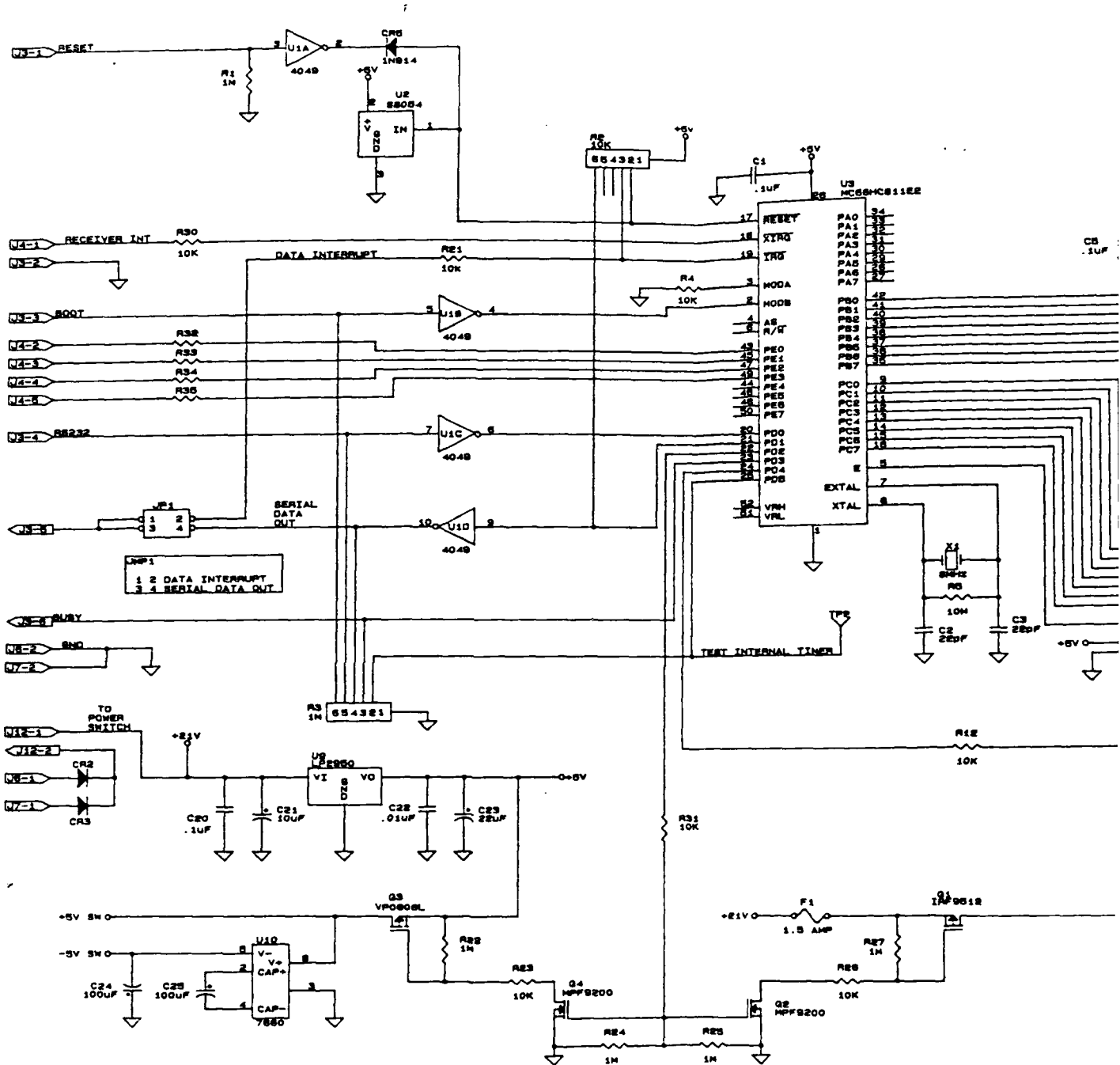


DATE	SPM	REVISION RECORD	AUTH	OR	CK

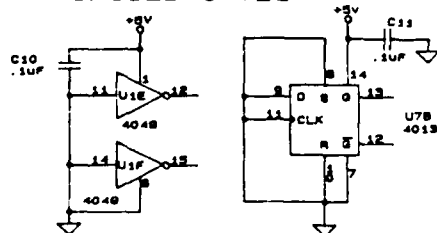


1-14-13 13-4-13

TOLERANCES EXCEPT AS NOTED		datasonics	
DECIMAL	SCALE	DRAWN BY	
FRACTIONAL	TITLE	APPROVED BY	
ANGULAR	DATE	DRAWING NUMBER	33



UNUSED GATES



NOTE:

R32-R35 ARE 10K 1/4 WATT

ALL RESISTORS IN OHMS

CR1, CR3 1N4004

TRANSMITTER PROCESSOR ASSEMBLY C630-03435

TRANSMITTER ASSEMBLY C630-03436

LAST USED

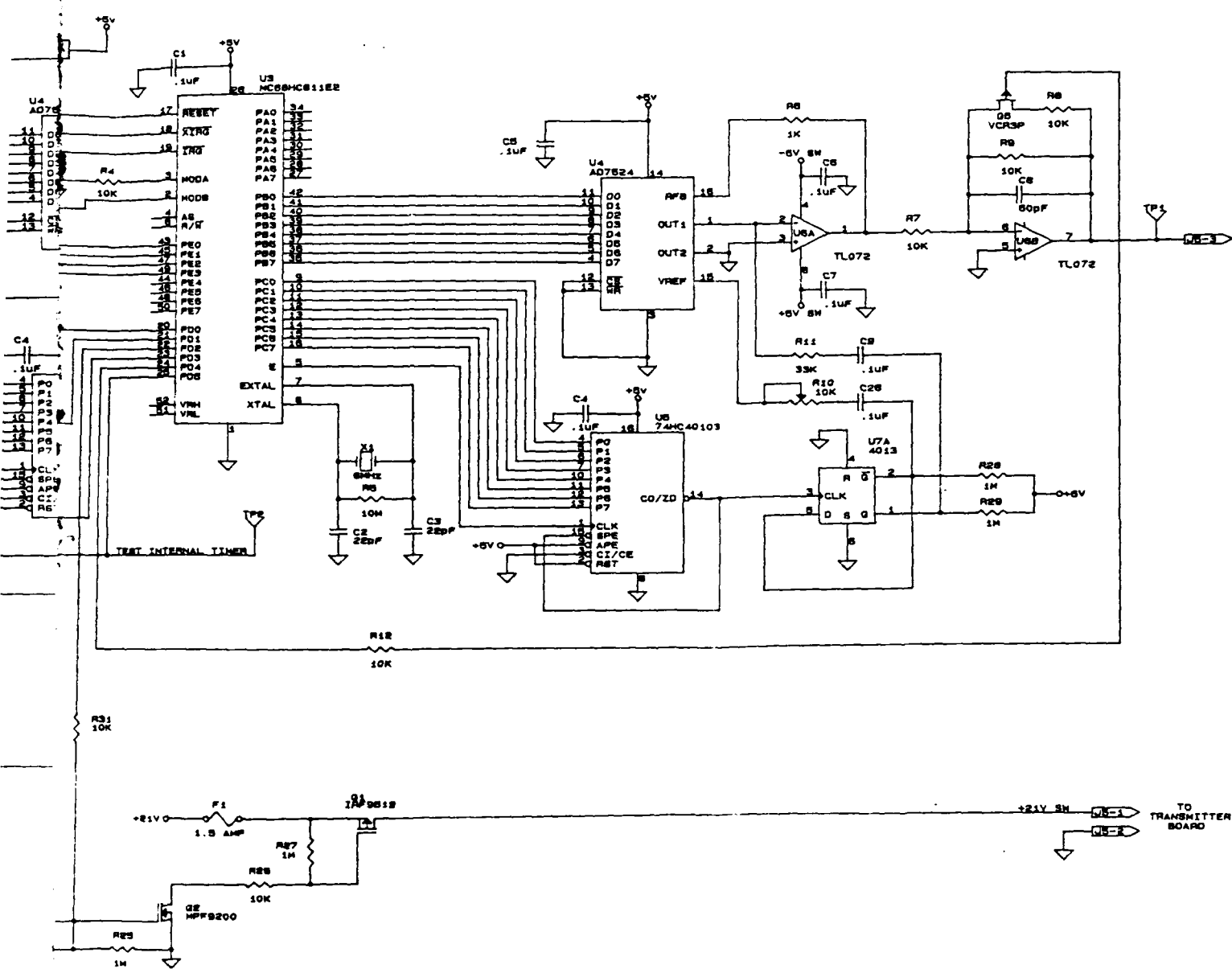
CR4

C28

U10

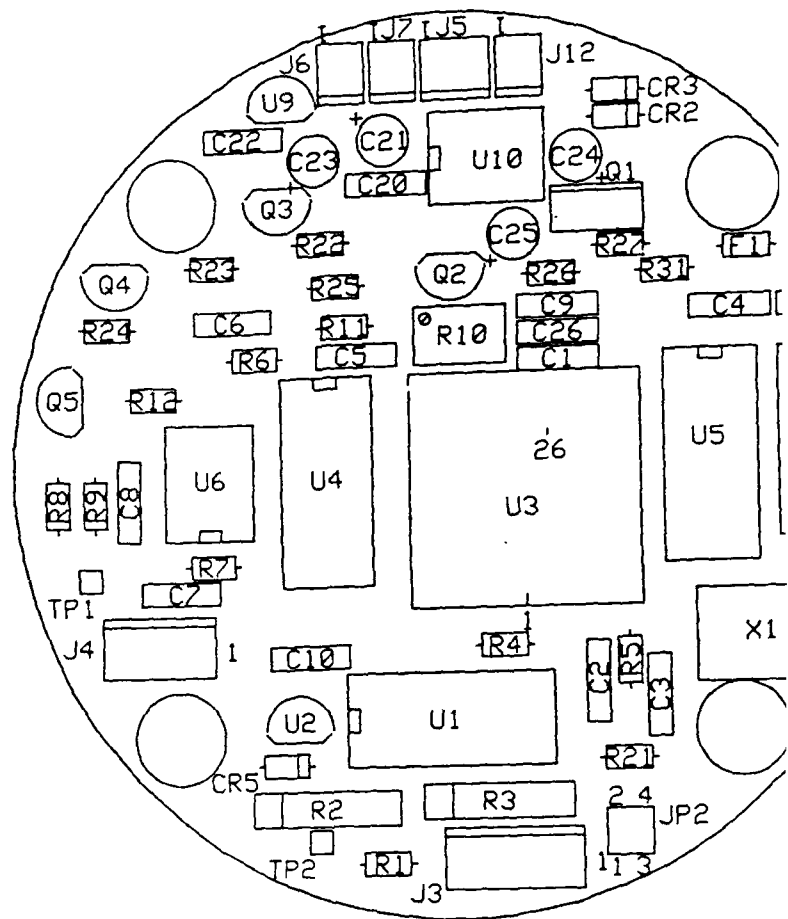
Q5

DATE	BY	REVISION RECORD	AUTH	DR	CR



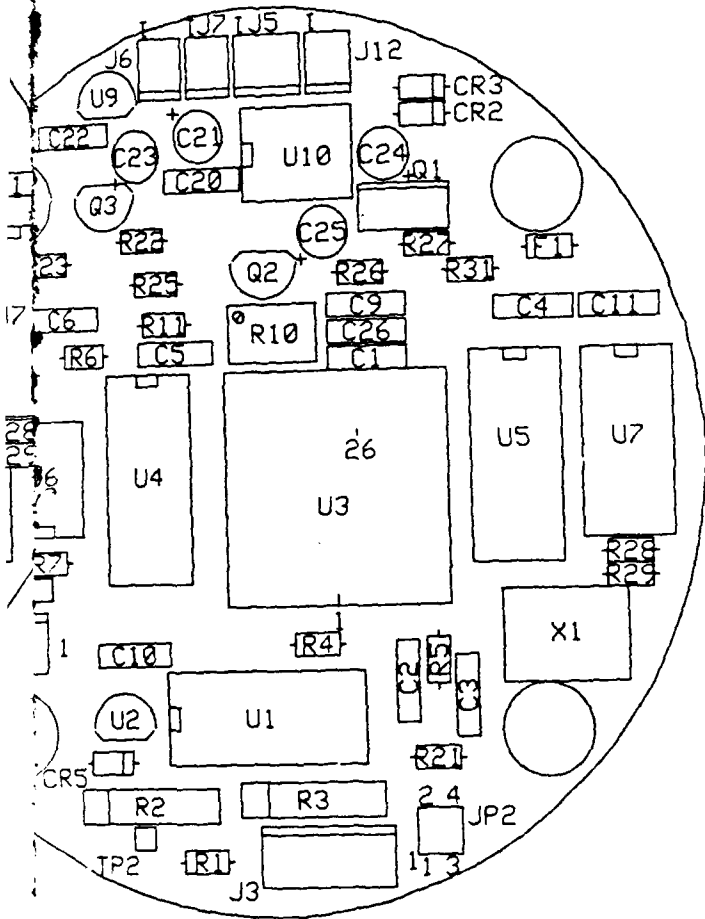
NOTE:
 R32-R36 ARE 10K 1/4 WATT
 ALL RESISTORS IN OHMS
 C12, C13 IN 4004
 TRANSMITTER PROCESSOR ASSEMBLY C830-03436
 TRANSMITTER ASSEMBLY C830-03438
 LAST USED
 CR4
 C29
 U10
 Q5

TOLERANCES (EXCEPT AS NOTED)		datasonics	
DECIMAL	SCALE	DRAWN BY	
1		APPROVED BY	
FRACTIONAL	TITLE	C830-03436	
2		DATE	
3		DRAWING NUMBER	
4		C830-03436	



SCHEMATIC 00000-03-10

DATE	TIME	REVISION RECORD	AUTH	DR	CR

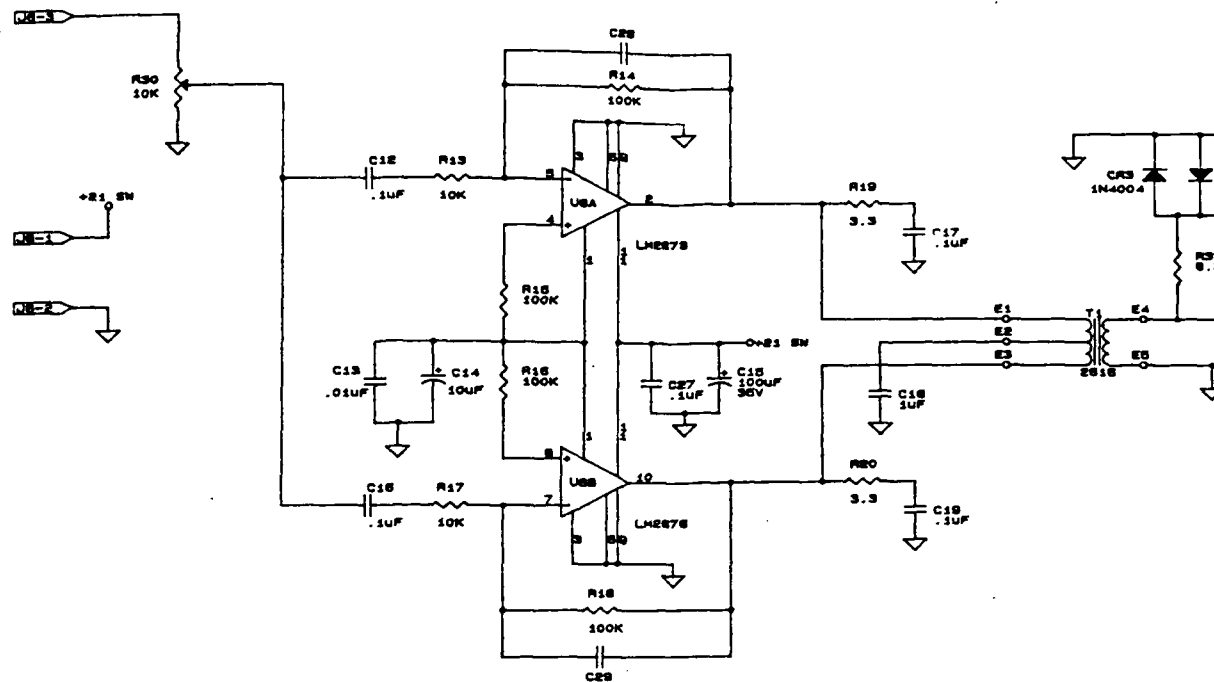


REVISED 06-10-63-1

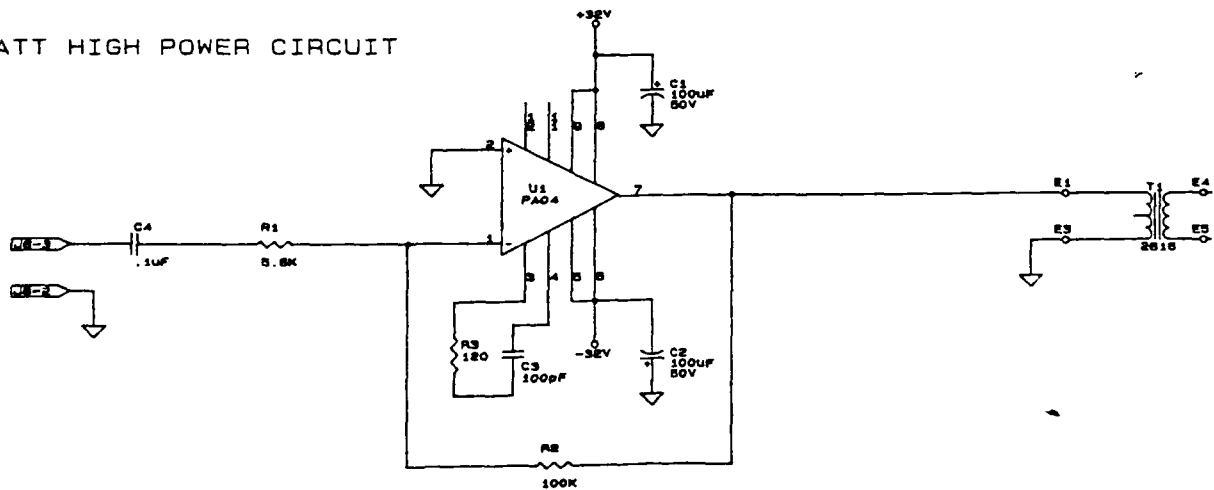
TOLERANCES (EXCEPT AS NOTED)		datasonics	
DECIMAL		SCALE	DRAWN BY
FRACTIONAL			APPROVED BY
ANGULAR		TITLE	
	DATE	DRAWING NUMBER	
		35	

NOTE:

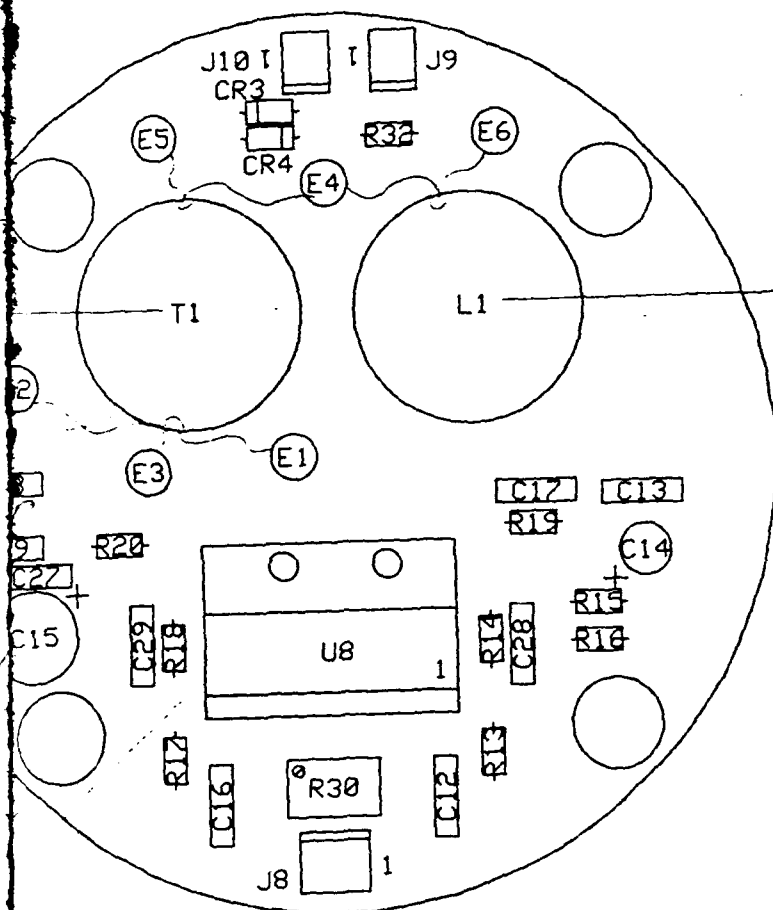
1. TRANSMITTER ASSEMBLY C630-03436
2. ALL RESISTORS IN OHMS
3. CR2, CR3 1N4004



100WATT HIGH POWER CIRCUIT



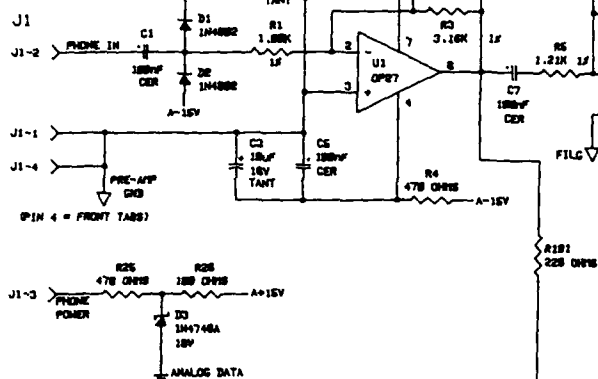
DATE	BY	REVISION RECORD	AUTH	OR	CR



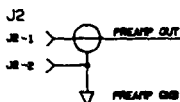
ASSEMBLY C230-03430

TOLERANCES (UNLESS OTHERWISE SPECIFIED)		datasonics	
DECIMAL		SCALE 2x	DRAWN BY JL
FRACTIONAL		APPROVED BY	
ANGULAR		TITLE ASSEMBLY TRANSMITTER	
DATE 1-6-64	DRAWING NUMBER C230-03430		37

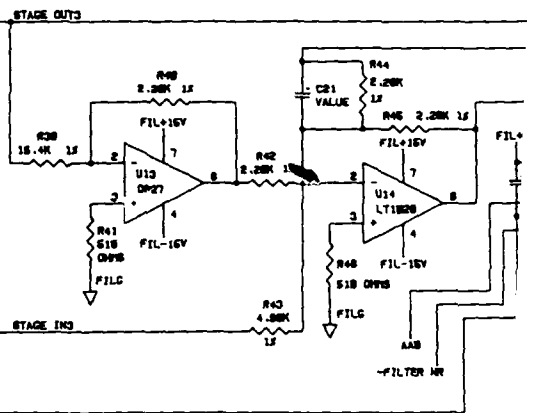
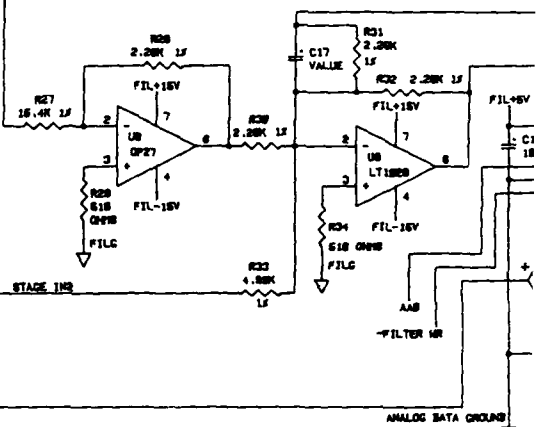
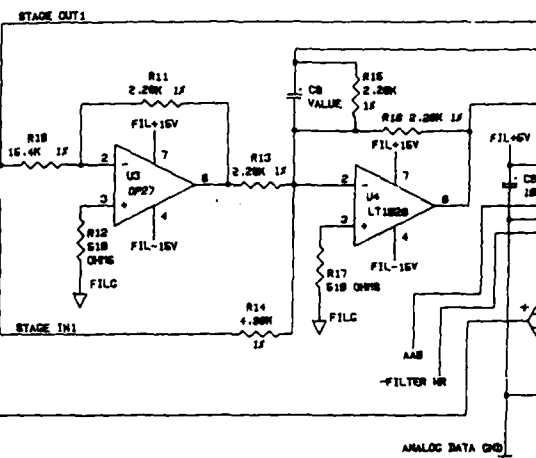
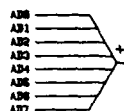
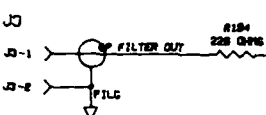
J1 = CIRCULAR 4 PIN DIN CONNECTOR



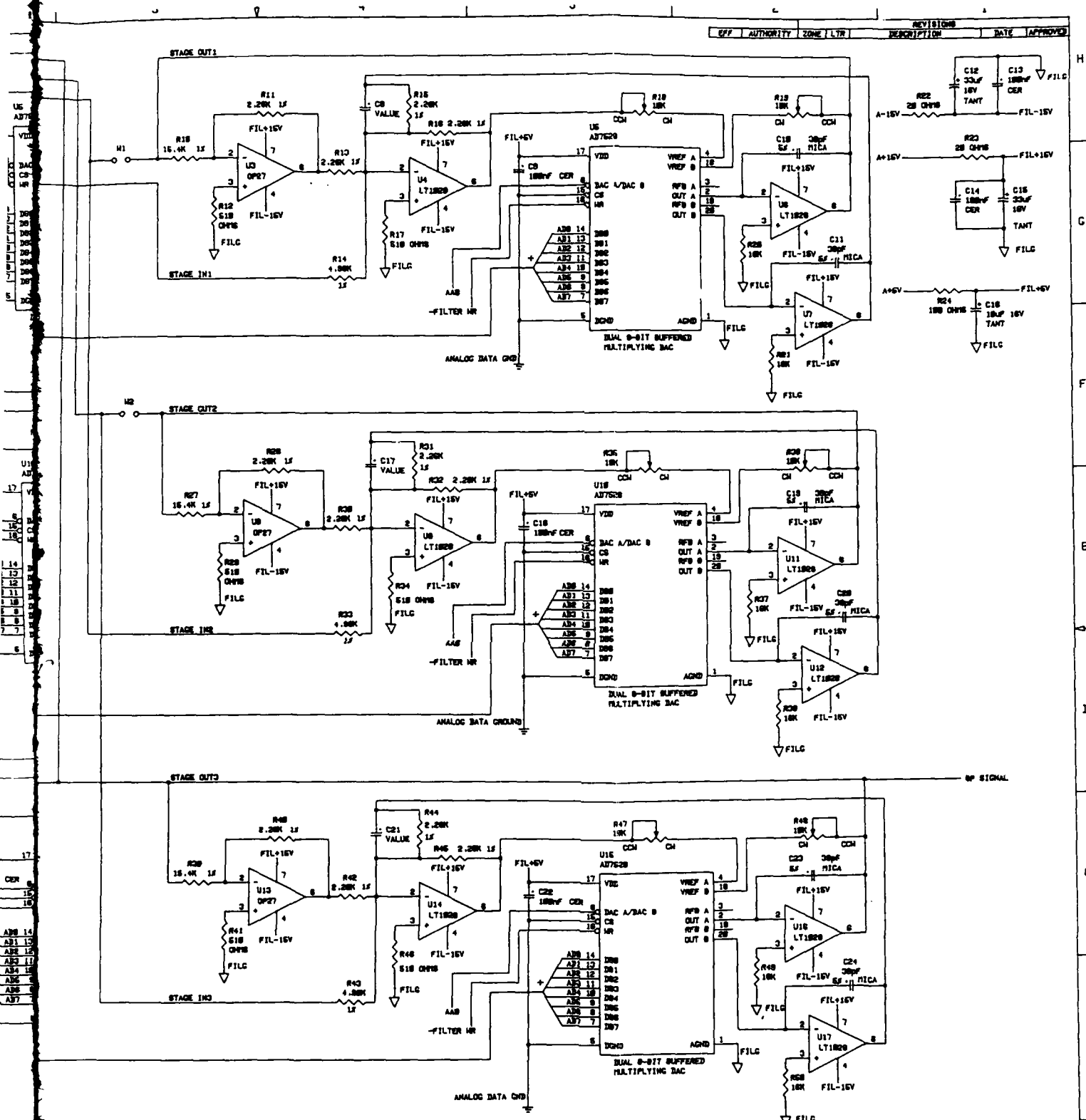
J2 = RIGHT ANGLE BNC



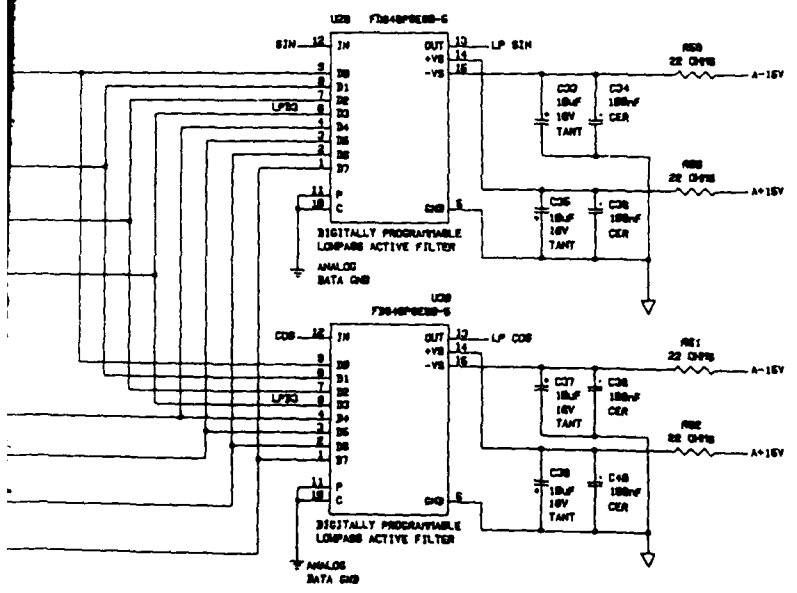
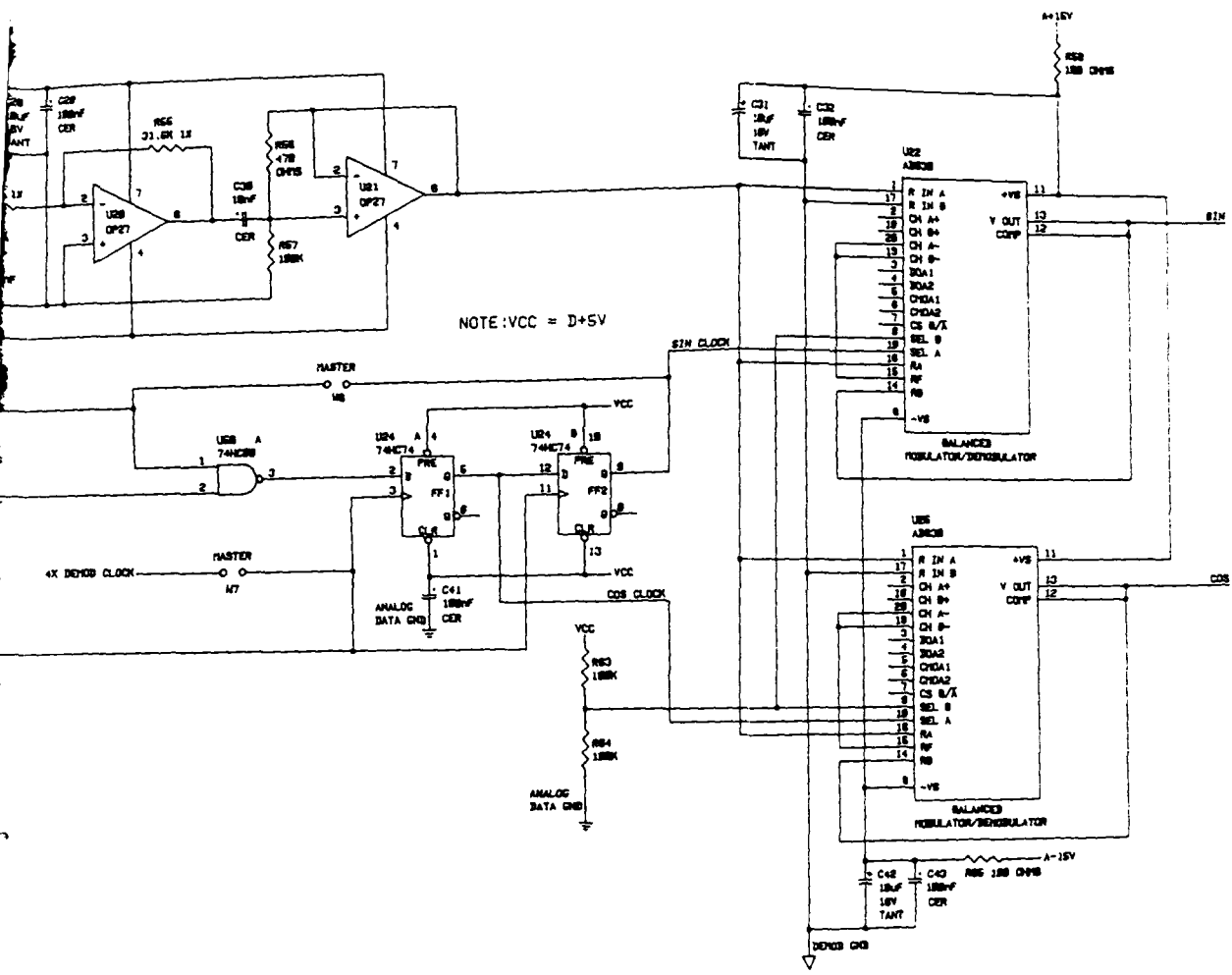
J3 = RIGHT ANGLE BNC



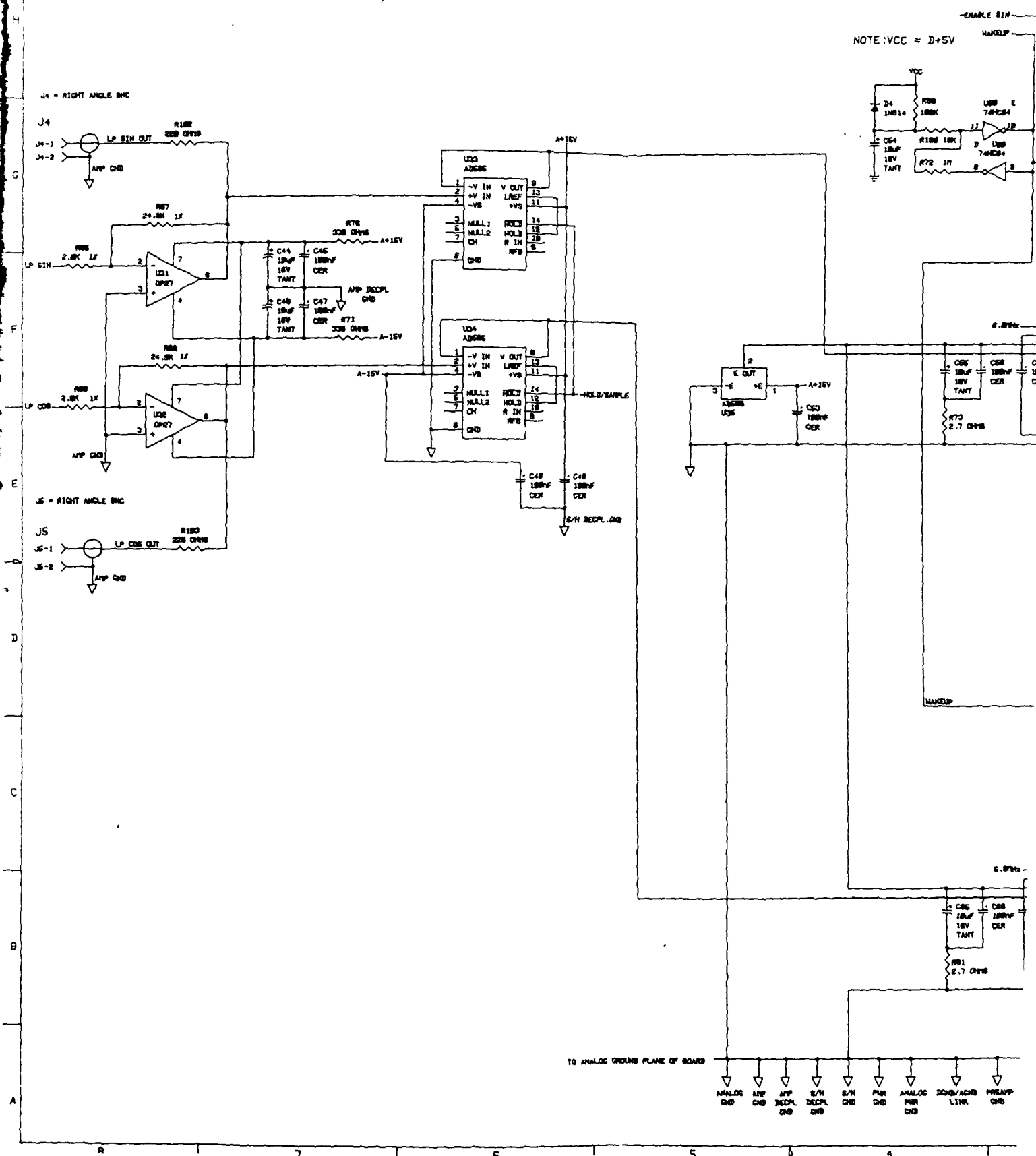
ANALOG DATA CH1

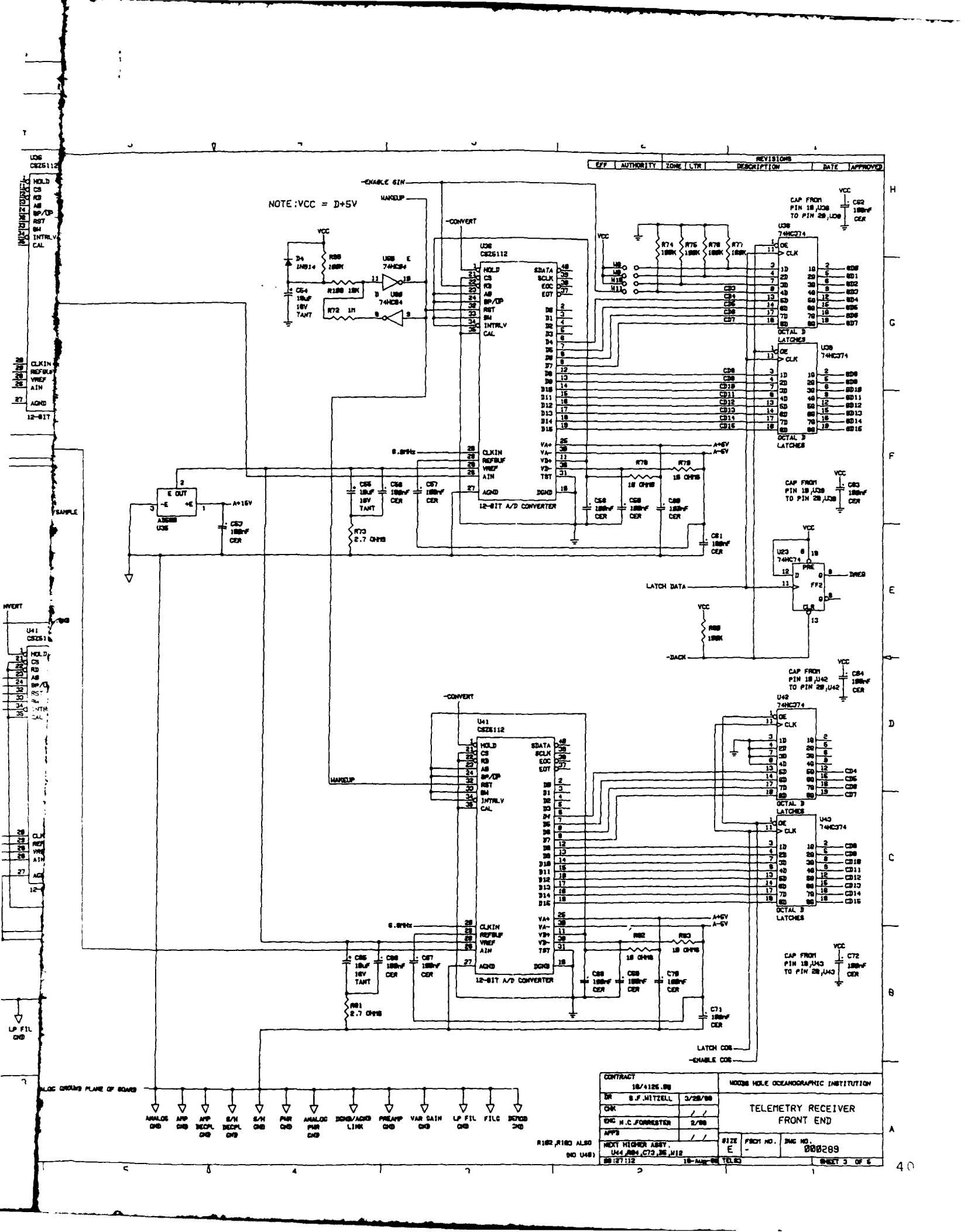


CONTRACT 16-4125-00		WOODS HOLE OCEANOGRAPHIC INSTITUTION	
DR S.F. MITZELL	2/14/80	TELEMETRY RECEIVER FRONT END	
CHK			
ENG H.C. FORRESTER	2/80	SIZE E	
APP			
NEXT HIGHER ASSY. U10, R51, C25, R4, U8		FIG. NO.	DWG. NO. 000289
16-4125-00		16-4125-00	SHEET 1 OF 5



CONTRACT 16-4125-00		NOORD NILE OCEANOGRAPHIC INSTITUTION	
DR S.F. MITCHELL	2/16/68	TELEMETRY RECEIVER FRONT END	
ENG H.C. FORMER	8/68		
APPS			
NEXT HIGHER ASSEY. U01,000,C44,04,05,06		SIZE E	FIG. NO. 000289
00111100	16-Aug-68	TELE	SHEET 2 OF 6



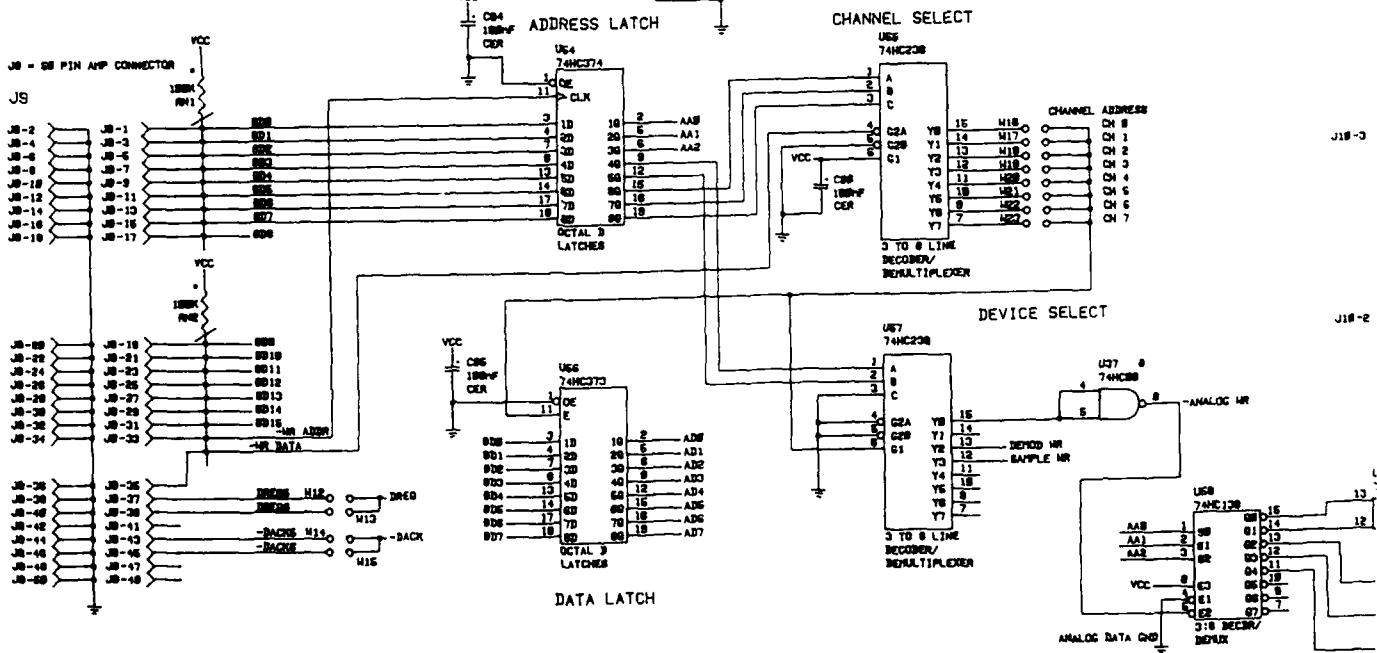
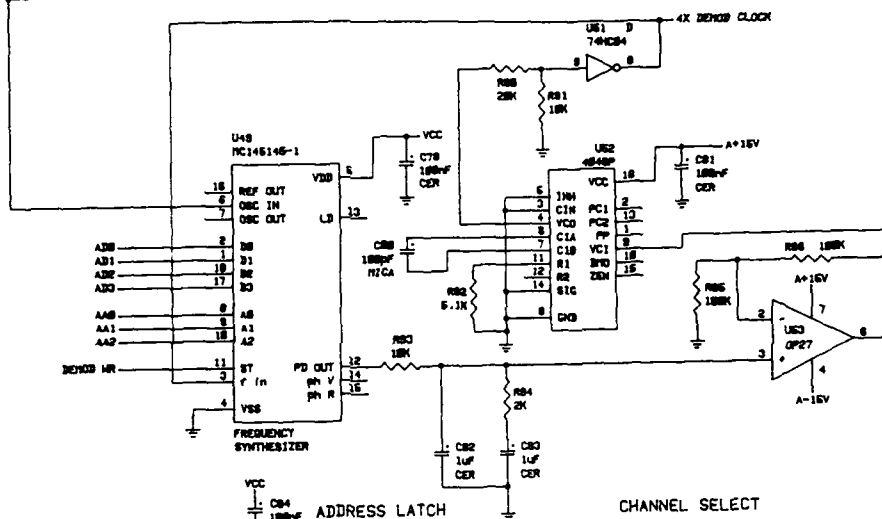
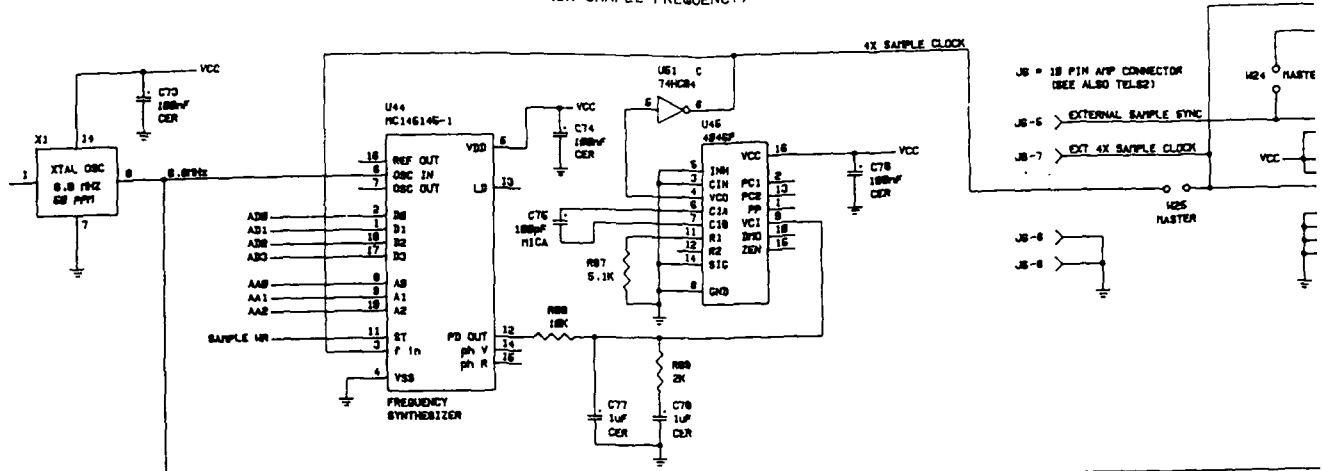


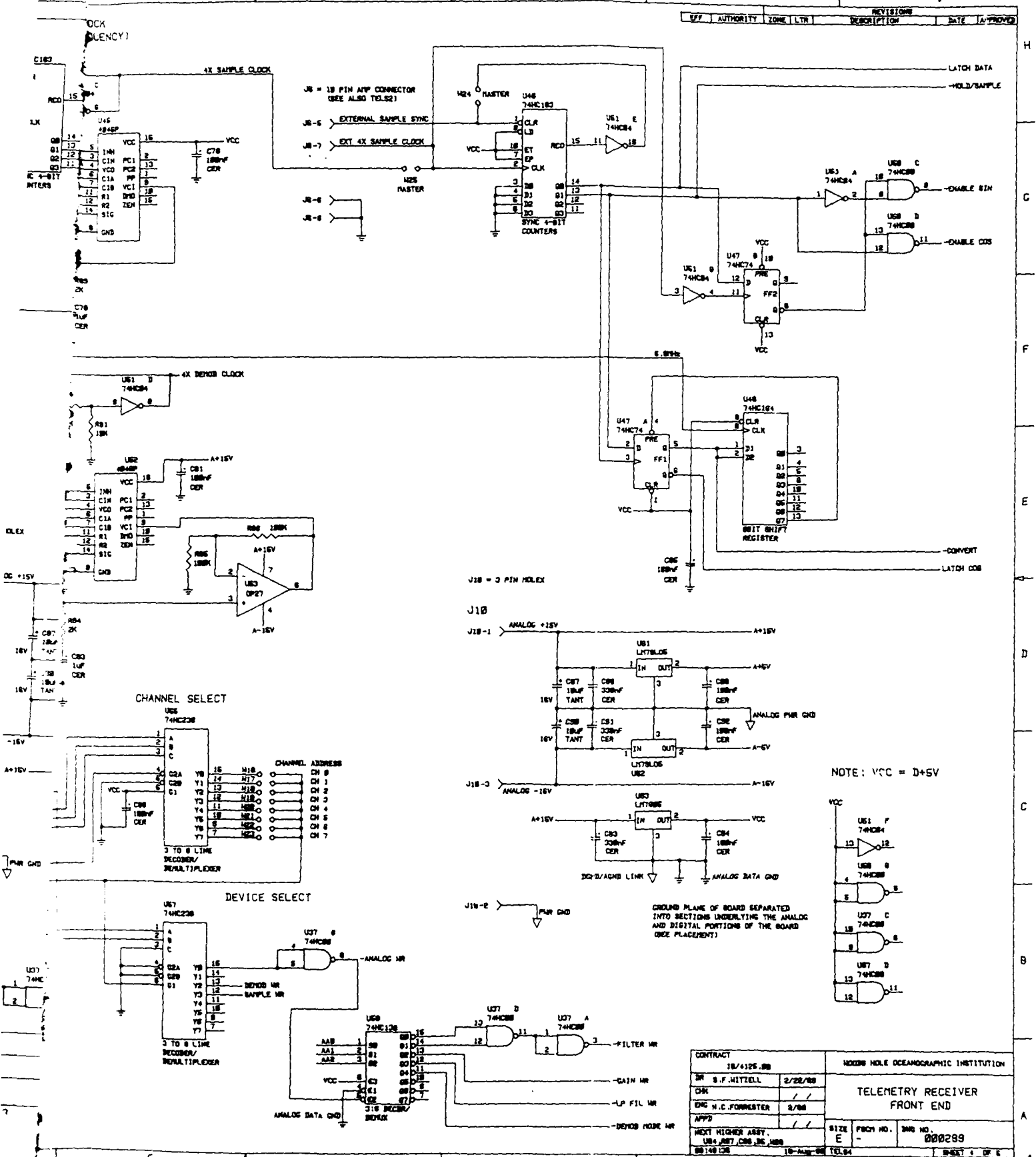
REV	AUTHORITY	ZONE	DATE	DESCRIPTION	APPROVED
-----	-----------	------	------	-------------	----------

CONTRACT		16/1125.00		WOODS HOLE OCEANOGRAPHIC INSTITUTION	
BY		S.F. MITZELL		3/28/80	
CHK		H.C. FORRESTER		2/80	
APP					
NEXT HIGHER ASSY.		U44, U45, C72, C73, U12		SIZE E	
BY		18-MAR-80		PCHD NO.	
18/1125		18-MAR-80		PAGE NO.	
				000289	
				SHEET 3 OF 6	

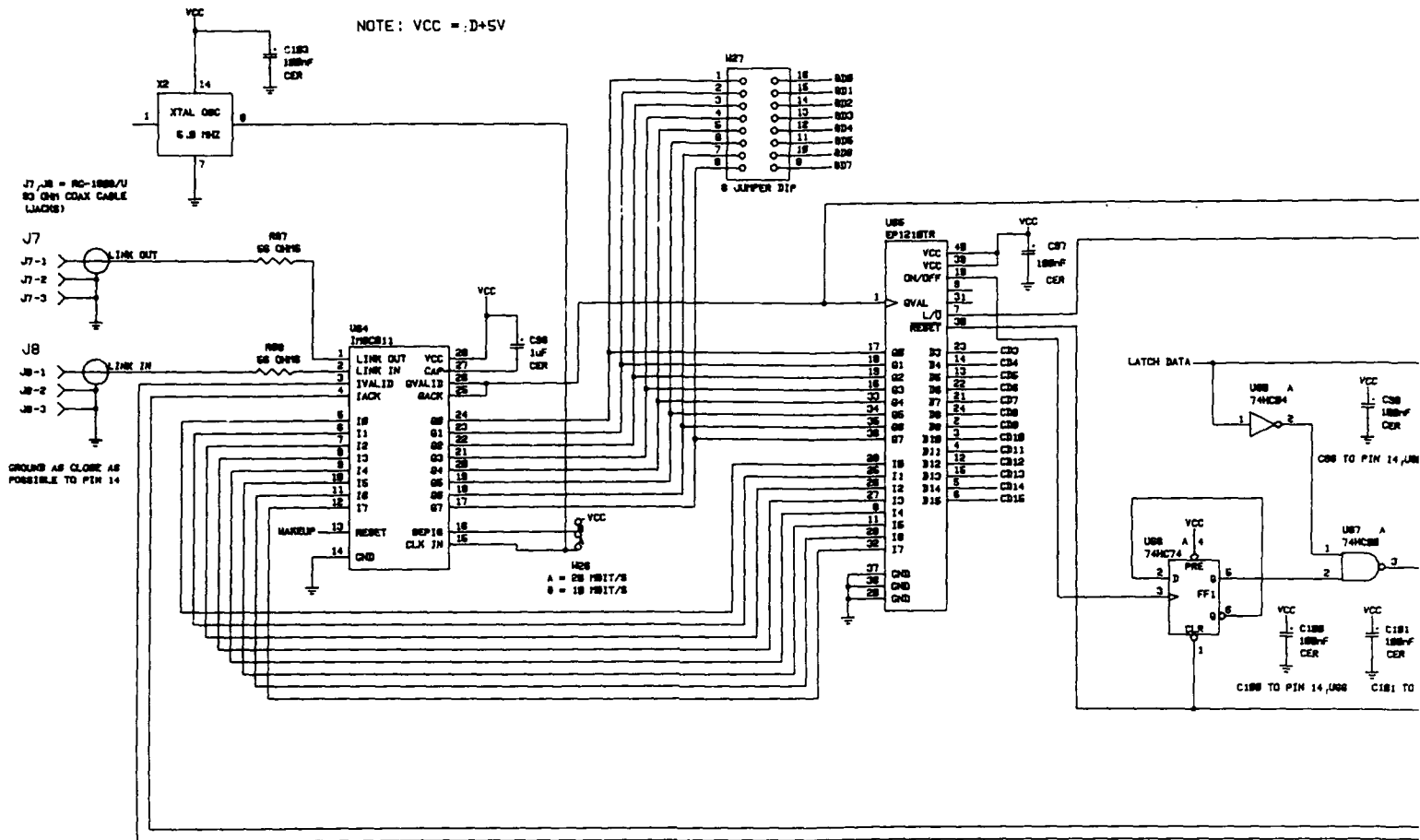
H
G
F
E
D
C
B
A

SAMPLE CLOCK (8X SAMPLE FREQUENCY)





NOTE: VCC = +5V



DOCUMENT LIBRARY

July 5, 1989

Distribution List for Technical Report Exchange

Attn: Stella Sanchez-Wade
Documents Section
Scripps Institution of Oceanography
Library, Mail Code C-075C
La Jolla, CA 92093

Hancock Library of Biology &
Oceanography
Alan Hancock Laboratory
University of Southern California
University Park
Los Angeles, CA 90089-0371

Gifts & Exchanges
Library
Bedford Institute of Oceanography
P.O. Box 1006
Dartmouth, NS, B2Y 4A2, CANADA

Office of the International
Ice Patrol
c/o Coast Guard R & D Center
Avery Point
Groton, CT 06340

Library
Physical Oceanographic Laboratory
Nova University
8000 N. Ocean Drive
Dania, FL 33304

NOAA/NESDIS Miami Library Center
4301 Rickenbacker Causeway
Miami, FL 33149

Library
Skidaway Institute of Oceanography
P.O. Box 13687
Savannah, GA 31416

Institute of Geophysics
University of Hawaii
Library Room 252
2525 Correa Road
Honolulu, HI 96822

Library
Chesapeake Bay Institute
4800 Atwell Road
Shady Side, MD 20876

MIT Libraries
Serial Journal Room 14E-210
Cambridge, MA 02139

Director, Ralph M. Parsons Laboratory
Room 48-311
MIT
Cambridge, MA 02139

Marine Resources Information Center
Building E38-320
MIT
Cambridge, MA 02139

Library
Lamont-Doherty Geological
Observatory
Columbia University
Palisades, NY 10964

Library
Serials Department
Oregon State University
Corvallis, OR 97331

Pell Marine Science Library
University of Rhode Island
Narragansett Bay Campus
Narragansett, RI 02882

Working Collection
Texas A&M University
Dept. of Oceanography
College Station, TX 77843

Library
Virginia Institute of Marine Science
Gloucester Point, VA 23062

Fisheries-Oceanography Library
151 Oceanography Teaching Bldg.
University of Washington
Seattle, WA 98195

Library
R.S.M.A.S.
University of Miami
4600 Rickenbacker Causeway
Miami, FL 33149

Maury Oceanographic Library
Naval Oceanographic Office
Stennis Space Center
NSTL, MS 39522-5001

Marine Sciences Collection
Mayaguez Campus Library
University of Puerto Rico
Mayaguez, Puerto Rico 00708